US006078974A

United States Patent [19]

Kirshtein

[11] **Patent Number: 6,078,974**

[45] **Date of Patent: Jun. 20, 2000**

[54] **METHOD AND APPARATUS FOR EXTENSION OF BI-DIRECTIONAL OPEN COLLECTOR SIGNALS IN A MULTIPLEXED DATA TRANSMISSION SYSTEM**

[75] Inventor: **Philip M. Kirshtein**, New Market, Ala.

[73] Assignee: **Cybex Computer Products Corporation**, Huntsville, Ala.

[21] Appl. No.: **09/057,002**

[22] Filed: **Apr. 8, 1998**

[51] **Int. Cl.**^{7} **G06F 15/16**

[52] **U.S. Cl.** **710/101**; 709/1; 709/227; 710/14; 710/65; 710/72; 710/73; 359/154

[58] **Field of Search** 710/65, 67, 14, 710/101, 72, 73; 709/1, 227; 341/177; 359/154

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,543,642	9/1985	Hansen	710/127
4,885,718	12/1989	Asprey et al.	710/101
5,457,784	10/1995	Wells et al.	710/9
5,587,824	12/1996	Asprey	359/154
5,602,595	2/1997	Citta et al.	348/495
5,615,393	3/1997	Kikinis et al.	710/67
5,621,734	4/1997	Mann et al.	709/227
5,815,109	9/1998	Tan	341/177
5,859,545	1/1999	Thornblad	326/90
5,870,395	2/1999	Baran	370/395
5,974,058	10/1999	Burns et al.	370/538
6,014,040	1/2000	Tracy	326/90

Primary Examiner—Thomas C. Lee
Assistant Examiner—Tammara Peyton
Attorney, Agent, or Firm—Nixon & Vanderhye P.C.

[57] **ABSTRACT**

An apparatus, method, and system for providing a virtual open collector circuit for a connection between a computer and computer peripherals, such as a mouse or keyboard, where peripheral data is transferred over a bidirectional communications system. The peripherals are located at distances from the computer greater than possible using typical computer cables. The apparatus has an arbitration means and a means for avoiding a lock-up condition.

11 Claims, 3 Drawing Sheets

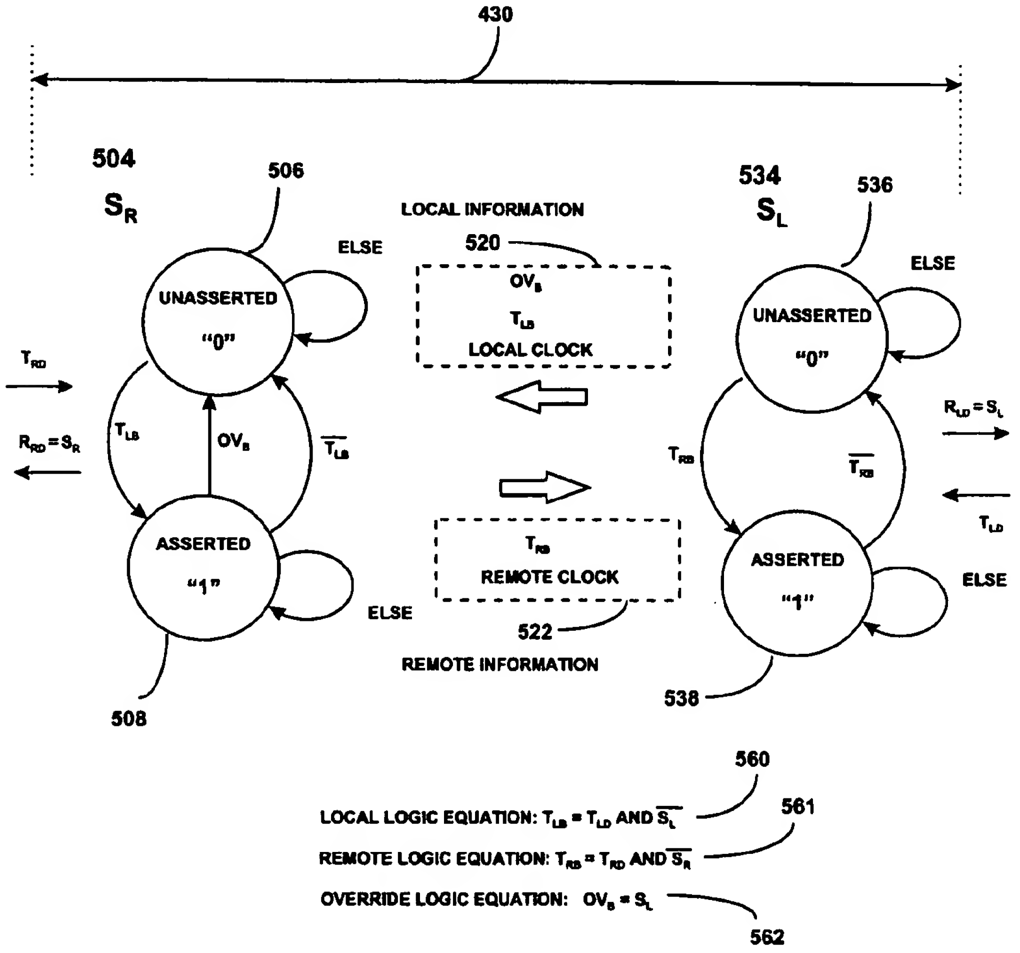


FIG.1

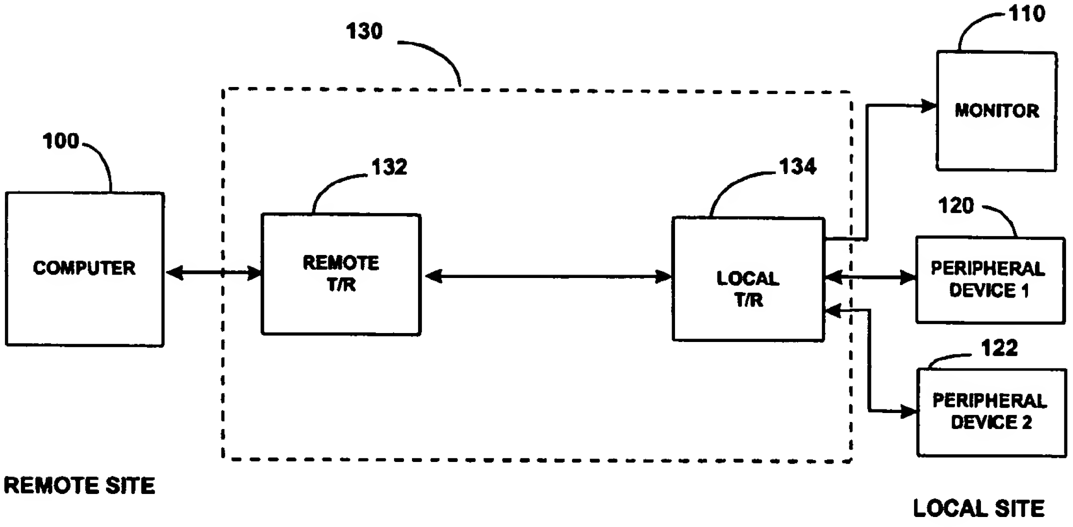


FIG.2

PRIOR ART

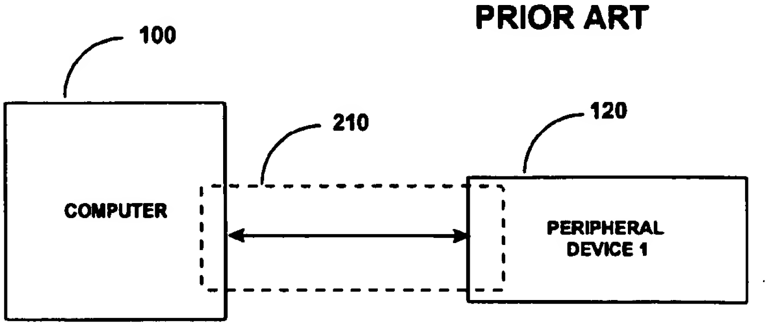


FIG.3

PRIOR ART

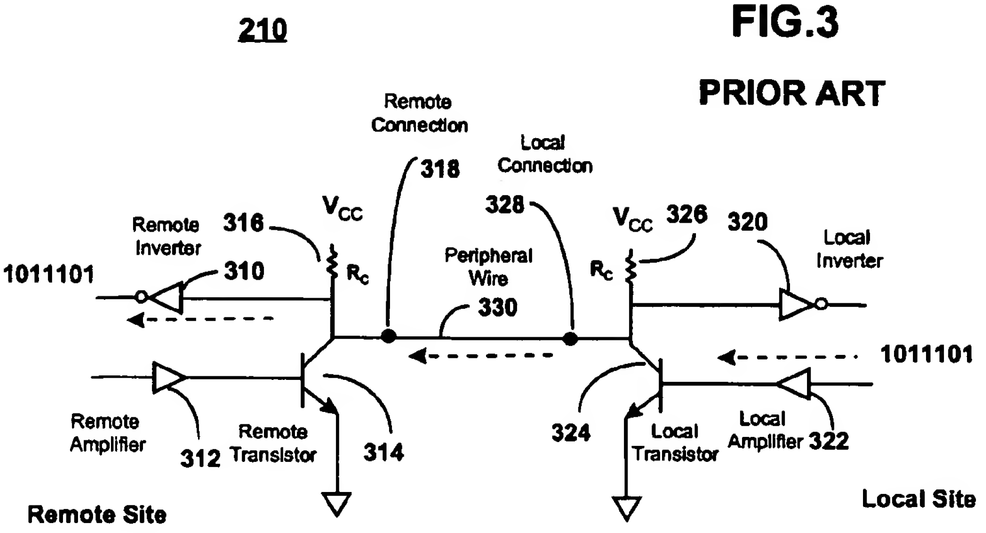


FIG.4

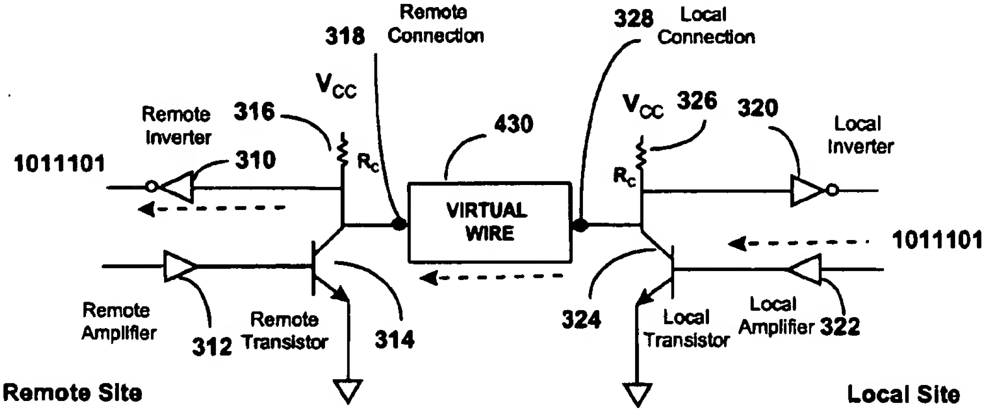
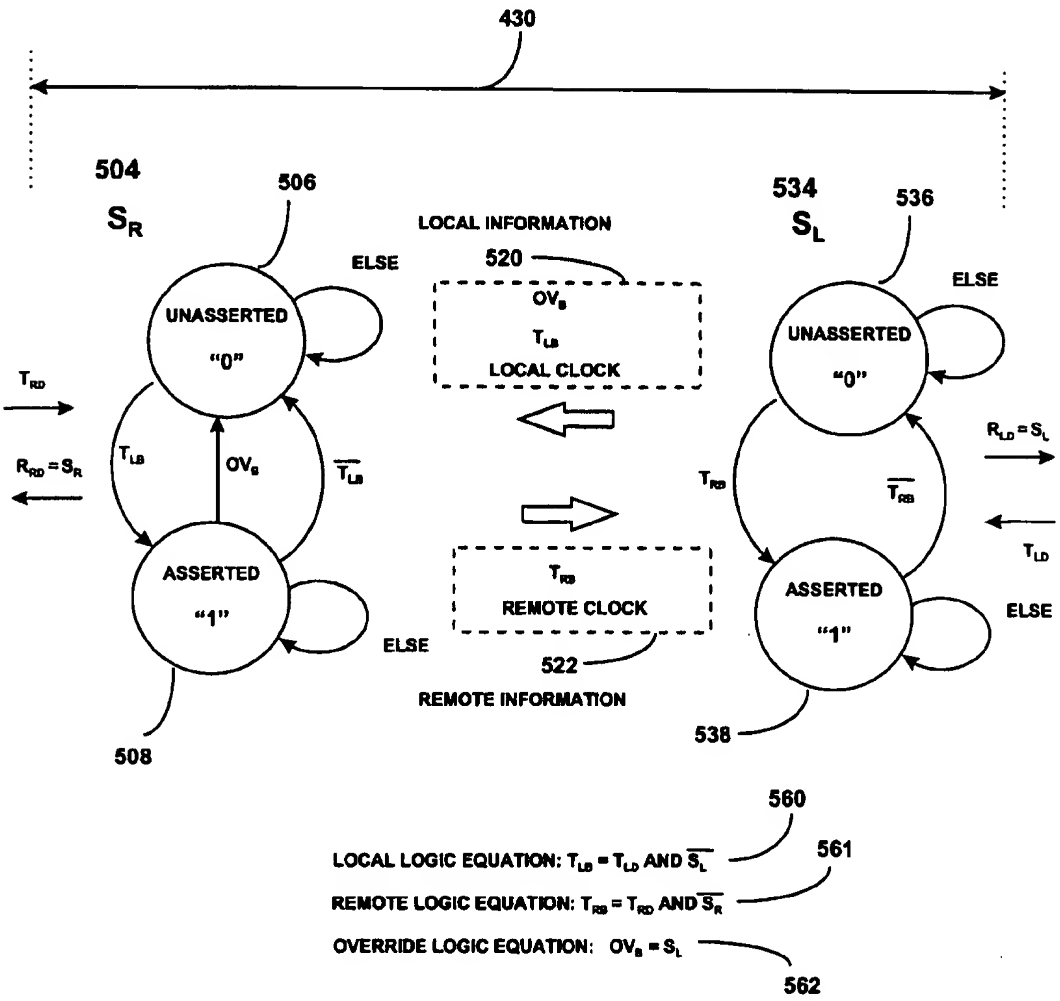


FIG.5


430
504
S_R
506
ELSE
UNASSERTED
"0"
T_{RD}
$R_{RD} = S_R$
T_{LB}
OV_B
$\overline{T_{LB}}$
ASSERTED
"1"
ELSE
508
LOCAL INFORMATION
520
OV_B
T_{LB}
LOCAL CLOCK
T_{RB}
REMOTE CLOCK
522
REMOTE INFORMATION
534
S_L
536
ELSE
UNASSERTED
"0"
$R_{LD} = S_L$
T_{RB}
$\overline{T_{RB}}$
T_{LD}
ASSERTED
"1"
ELSE
538
560
561
LOCAL LOGIC EQUATION: $T_{LB} = T_{LD}$ AND $\overline{S_L}$
REMOTE LOGIC EQUATION: $T_{RB} = T_{RD}$ AND $\overline{S_R}$
OVERRIDE LOGIC EQUATION: $OV_B = S_L$
562

METHOD AND APPARATUS FOR EXTENSION OF BI-DIRECTIONAL OPEN COLLECTOR SIGNALS IN A MULTIPLEXED DATA TRANSMISSION SYSTEM

FIELD OF THE INVENTION

This application relates to an extension system for open collector circuits used for a connection between a computer and computer peripherals such as a mouse or keyboard. When the computer is located at a first site and the peripherals at a second site the extension system provides a way to transfer data in a manner transparent to the user and has a means for avoiding a lock-up condition which may occur when bidirectional data is transmitted over an extension system.

BACKGROUND OF THE INVENTION

A personal computer is typically coupled to peripherals that serve as computer-user interfaces. Such peripherals may include a keyboard, a mouse, and a monitor. Typically standard cables connecting each peripheral are around four feet long. Because of the characteristics of the connecting cables and their interfaces, the cables do not provide a reliable peripheral connection when they are much longer than twenty feet.

There are situations where it is desirable to separate the computer from the peripherals at distances much greater than allowed by typical standard peripheral cables. The apparatus to extend the distance between a computer and peripherals is called an extender or an extension system. One such system, PC-Extender Plus is available from Cybex Computer Products of Huntsville, Ala. The PC-Extender Plus allows a keyboard, monitor, and mouse to be placed up to 600 feet away from the computer. An extender typically has two electronic boxes and several cables between the computer and the peripherals. As the demand for extenders increases, the need for more efficient use of cables has developed.

In existing extenders, standard coaxial cables, shielded cables, and unshielded cables serve as communication channels. In addition, custom cables have been developed to provide an improved communication channel for some extender systems. The cables connecting the computer to the peripherals may be confined in a binder or may be separate cables for each of the desired connections. The apparatus or electronic boxes connected to the ends of each cable serve as interfaces between the computer and the cable at one end and the peripherals and the cable at the other end.

A recently developed digital system for extending computer peripherals is described in U.S. application Ser. No. 08/971,223 filed on Nov. 15, 1997 still pending and is assigned to the assignee of the present invention and is hereby incorporated as a reference herein. One of the problems occurring in the recently developed system was the coupling of open collector circuits between the computer and peripherals in a way that was compatible with existing hardware and operationally transparent to a computer user. A similar problem occurs in analog extension systems and was solved as described in U.S. Pat. No. 5,587,824 issued to Asprey. In Asprey an arrangement of analog circuit elements is used to allow a long wire to appear as a short wire to open collector circuits. The recently developed system of application '223 provides a bidirectional data flow using a half duplex transmission method and placing peripheral bits in data packets. Because the recently developed digital system converts analog signals to digital signals, transmits the digital signals in packets, then converts the digital signals back to analog signals the invention of Asprey is not applicable.

Accordingly, it is the object of this invention to provide a means for coupling open circuit collector circuits in a manner that is transparent to the user and does not add significant cost to a digitally based extension product. To provide such a means requires the development of an algorithm, method, or apparatus within the framework of the recently developed system or other bidirectional data communication system. Such a solution should function as a virtual wire to the user and perform essentially as a short piece of wire typically used to couple open collector circuits.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is a block diagram illustrating a computer coupled to peripherals via an extension system in accordance with the present invention.

FIG. **2** is a block diagram illustrating a prior art coupling of a peripheral device to a computer.

FIG. **3** is a schematic illustrating a prior art connection.

FIG. **4** is a schematic illustrating an embodiment of the present invention.

FIG. **5** is a state diagram illustrating an embodiment of the present invention.

DESCRIPTION OF THE PREFERRED EMBODIMENT

FIG. **1** illustrates a computer **100** coupled to a monitor **110**, a first peripheral **120**, and a second peripheral **122**. A bidirectional data transmission system **130** having a remote transmitter/receiver **132** and a local transmitter/receiver **134** provides an information path for the computer **100** to exchange information with computer elements. The bi-directional data flowing between the remote transmitter/receiver **132** and the local transmitter/receiver **134** are placed in high-speed packets. Details of one such bi-directional data transmission system is described in the above mentioned U.S. application Ser. No. 08/971,223 assigned to the assignee of the present invention. The bidirectional data transmission system described in '223 is a half-duplex transmission system. Because other data transmission systems, such as a full-duplex transmission system, may serve as a communication channel for the present invention the use of a half-duplex transmission system is not a limitation.

A block diagram illustrating a prior art computer to peripheral coupling is shown in FIG. **2**. The computer **100** and the first peripheral device **120** exchange bidirectional information using prior art coupling link **210**. The prior art coupling link comprises a circuit in the computer, a piece of cable, and a circuit in the peripheral device. The first peripheral device may be a computer mouse and the piece of cable a wire in the cord connecting the computer mouse to the computer. The first peripheral device may also be a keyboard or similar device that use an interface circuitry contained in a mouse.

Referring now to FIG. **3**, a prior art circuit, there is shown a peripheral cable **330** connected on one end to the collector of a remote transistor **314** and connected on the other end to the collector of a local transistor **324**. Also shown and functionally arranged are elements known to those skilled in the art including a voltage supply, collector resistors **316**, **326**, inverters **310**, **320**, and amplifiers **312**, **322**. In the prior art circuit, data may be transferred bi-directionally.

A serial data stream is shown, represented as binary ones and zeros, entering the local amplifier **322**. When a binary one is asserted and is applied to the local amplifier the base of the local transistor is biased and pulls the collector to ground potential. The value of voltage at the local connection point **328**, on the peripheral cable **330**, and at the remote connection **318** is essentially at ground potential. Remote inverter **310** converts the ground potential level at remote connection point **318** to an output voltage representing a binary one. When the next value of the serial data stream, now a binary zero, is applied to the local amplifier **322**, the local transistor **324** is biased to an off condition. The voltage at the local connection **328**, on the peripheral cable **330**, and at the remote connection **318** moves from the ground potential representing a binary zero to approximately V_{CC} representing a binary one. The remote inverter **310** converts the binary one to a binary zero. As the off-on process continues at the local transistor **324** in response to the transmitted local serial data, the output of the remote inverter **310** becomes remote received serial data having the same bit pattern as the input to the local amplifier **322**. In summary, data is transferred from the remote site to the local site using remote amplifier **312**, remote transistor **314**, the peripheral cable **330**, and the local inverter **320**. The peripheral wire **330** is a piece of wire contained in a peripheral cable connecting a peripheral to a computer. Those familiar with computers would recognize, for example, the piece of wire as one of the wires in a mouse cable or a keyboard cable.

FIG. **4** illustrates a virtual wire **430** serving as the peripheral wire **330** for coupling circuit elements at the remote site to the local site. The virtual wire of the present invention uses a bidirectional data communication system as a link between local processing elements and remote processing elements. The processing elements include state machines and logic equations plus interface circuits to provide the appropriate voltages at the remote connection **318** and the local connection **328**. The state machines and logic equations of the present invention may be implemented using discrete logic, a programmed logic array, a programmed microprocessor, or similar devices. The interface circuits are typically comprised of amplifiers and inverters and well known by those skilled in the art. The state machines and logic equations serve as a means for transmission, arbitration, and a means for avoiding a latch-up condition and are essential elements of the present invention.

To assist with an understanding of the present invention, as illustrated in FIG. **5**, the following notation is used:

S_R-state of the remote state machine;

S_L-state of the local state machine;

R_{RD}-received remote data at the remote site;

R_{LD}-received local data at the local site;

T_{RD}-transmitted remote data from the remote site;

T_{LD}-transmitted local data from the local site;

T_{RB}-transmitted remote bit from the remote site;

T_{LB}-transmitted local bit from the local site; and

OV_B-override bit.

Referring now to FIG. **5** there is shown a local state machine **534** and a remote state machine **504**. The remote state machine **504** has an asserted state **508**, binary one, and an unasserted state **506**, binary zero and the local state machine **534** has an asserted state **538**, binary one, and an unasserted state **536**, binary zero. The local state machine goes from the unasserted state **536** to the asserted state when T_{RB} is asserted, else remains in the unasserted state. The local state machine **534** goes from the asserted state **538** to the unasserted state when T_{RB} is unasserted else remains in the asserted state. Similar state changes occur in the remote state machine as shown in FIG. **5**. However the remote state machine has an additional means of changing states, when the override bit, OV_B, is asserted the remote state machine transfers from the asserted state **508** to the unasserted state **506**. The override bit serves to keep the system from having a lock-up condition due to a positive feedback relation between the two state machines.

Initially, when no data has been transferred for a period of time, both state machines are in the unasserted state. When data is sent from the local site T_{LD} is a sequence of bits. The first bit of the sequence, assumed to be a binary one, is converted to a transmitted local bit, T_{LB}, according to the local logic equation **560**. The LOCAL CLOCK and T_{LB} are transmitted to the remote state machine **504** over the bidirectional transmission system **130**. The remote state machine **506**, being in the unasserted state **506**, then changes to the asserted state **508** and the remote received data has a first bit in an output sequence equal to binary one. When the second bit of T_{LD}, which may be a binary zero or a binary one, is processed then the local logic equation again generates a new transmitted local bit. The new transmitted local bit becomes the second bit of the remote received data in accordance with the operation of the local logic equation **560** and the remote state machine **504**. When all the transmitted local data has been transferred both state machines return to an unasserted state.

Transmitted remote data, T_{RD}, is transferred to the local site over the bidirectional transmission system using remote logic equation **561** and the local state machine **534**. When the transmitted remote bit and REMOTE CLOCK arrive at the local site, the local state machine goes to the state as shown in FIG. **5**. Received local data appears as a sequence of bits at the local site.

In summary, the state machines of FIG. **5** are initially in an unasserted state. If local data is transmitted from the local site to the remote site the state of the local state machine and the values of the data, according to the local logic equation **560**, determine the value of the bit transmitted over the bidirectional transmission system to the remote state machine **504**. The local logic equation and remote state machine process one bit at a time sequentially generating received remote data at the remote site. Data is transferred from the remote site to the local site in a similar manner. When interface circuits at the local and remote sites assure the outputs are coupled at the correct voltage level to the collectors of the remote transistor **314** and the local transistor **324**, the state machines and logic equations serve as the virtual wire **430**.

From the foregoing, it will be observed that numerous variations and modifications may be effected without departing from the spirit and scope of the novel concept of the invention. It is to be understood that no limitation with respect to the specific methods and apparatus illustrated herein is intended or should be inferred. It is, of course, intended to cover by the appended claims all such modifications as fall within the scope of the claims. The invention is further defined by the following claims:

We claim:

1. In an extension system for transferring local data from a local site to a remote site using a virtual open collector circuit between the local site and the remote site, the method comprising the steps of:

sampling a data bit at the local site at a local clock rate into a stream of local samples corresponding to the data bit;

at each sample clock, determining a local state condition S_L of the site; creating a local transmission bit for each local sample as a function of both the local sample and S_L;

transmitting the local transmission bit to the remote site;

determining whether to reset a state condition of the remote site to avoid a bi-stable latch-up condition of the virtual open collector circuit; and

repeating the generating, determining, creating, and transmitting steps for data bits at the remote site sampled into remote samples at a remote clock rate and converted into remote transmission bits as a function of both a remote state condition and the remote sample.

2. The method of claim **1** wherein the local data is from a computer peripheral.

3. An extension system for transferring local data from a local site to a remote site using a virtual open collector circuit between the local site and the remote site, comprising:

a generator at the local site for receiving a data bit, sampling the data bit into a stream of local samples, and determining a local state condition of the local site and transmitting to the remote site a combined signal each being a combination of a current local sample with a current local state condition; and

a remote state machine at the remote site receiving the combined signal and the local state condition and therefrom decoding the data bit and determining whether to reset a remote state condition of the remote state machine to avoid a bi-stable latchup condition of the virtual wire.

4. The method of claim **3** wherein the local data is from a computer peripheral.

5. In an extension system for transferring data between a local site and a remote site where the data is transferred over a bidirectional data transmission system, an apparatus for providing a virtual open collector circuit between the local site and the remote site, the apparatus comprising:

a local generator at the local site for generating a stream of local data bits and creating a logic state local signal for each local data bit to satisfy the equation $T_{LB}=T_{LD}$ AND S_L where T_{LB} is the created local logic state signal, T_{LD} is a current local data bit, and S_L is a current state condition of a local state machine;

a remote state machine at the remote site receiving T_{LB} and S_L from the local site, and decoding the stream of local data bits from T_{LB} and making a reset decision of a state condition of the remote state machine based on S_L;

a remote generator at the remote site for generating a stream of remote data bits and creating a logic state remote signal for each remote data bit to satisfy the equation $T_{RB}=T_{RD}$ AND S_R, where T_{RB} is the created remote logic state signal, T_{RD} is a current remote data bit, and S_R is a current remote condition of the remote state machine; and

the local state machine at the local site receiving T_{RB} and S_R from the remote site, and decoding the stream of remote data bits from T_{RB} and making a reset decision of a state condition of the local state machine based on S_R.

6. The method of claim **5** wherein data from the local site is generated by a computer peripheral and data from the remote site is generated by a computer.

7. The method of claim **6** wherein the computer peripheral is a mouse.

8. The method of claim **5** wherein the local generator further generates an override bit where the override bit serves as a means for avoiding a latch-up condition.

9. A virtual wire communication link between a computer and a peripheral normally associated with the computer, comprising:

a master state machine bi-directionally communicating with the peripheral via a first link, oversampling a logic state of the first link and providing a master combined result of a combination of the oversampling with a determined master state condition;

a slave state machine, communicating with the master state machine via a dedicated channel and bi-directionally communicating with the computer via a second link, for receiving the master combined result from the master state machine via the dedicated channel, mimicking the logic state of the first link based on the master combined result, and driving the second link to the computer with the mimicked logic state of the first link;

the slave state machine also oversampling a logic state of the second link and providing a slave combined result of a combination of the oversampling with a determined slave state condition;

the master state machine also receiving the slave combined result from the slave state machine via the dedicated channel, mimicking the logic state of the second link based on the slave combined result and driving the first link to the peripheral with the mimicked logic state of the second link.

10. A virtual wire communication system as in claim **9**, wherein:

the master state machine also periodically sends in addition to the master combined result the master state condition to the slave machine via the dedicated channel; and the slave state machine receives the master state condition and based on the master state condition resets its own state condition to avoid a bi-stable latch up of the virtual wire.

11. A virtual wire communication system as in claim **1**, wherein the local transmission bit satisfies the logic equation $T_{LB}=T_{LD}$ AND S_L, where T_{LB} is the local transmission bit and T_{LD} is the local sample.

* * * * *

US006112264A

United States Patent [19]

Beasley et al.

[11] **Patent Number: 6,112,264**

[45] **Date of Patent: Aug. 29, 2000**

[54] **COMPUTER INTERCONNECTION SYSTEM HAVING ANALOG OVERLAY FOR REMOTE CONTROL OF THE INTERCONNECTION SWITCH**

[75] Inventors: **Danny L. Beasley**, Mukilteo; **Robert V. Seifert, Jr.**, Redmond; **Paul Lacrampe**, Seattle; **James C. Huffington**, Edmond; **Thomas Greene**, Bellevue; **Kevin J. Hafer**, Woodinville, all of Wash.

[73] Assignee: **Apex PC Solutions Inc.**, Woodinville, Wash.

[21] Appl. No.: **09/244,947**

[22] Filed: **Feb. 4, 1999**

Related U.S. Application Data

[63] Continuation of application No. 08/969,723, Nov. 12, 1997, Pat. No. 5,884,096, which is a continuation of application No. 08/519,193, Aug. 25, 1995, Pat. No. 5,721,842.

[51] **Int. Cl.**^{7} **G06F 13/14**; G06F 3/153

[52] **U.S. Cl.** **710/38**; 345/113; 345/327; 348/589

[58] **Field of Search** 345/113, 326, 345/327, 329, 1, 2, 4, 7, 9; 710/15, 38, 2, 51, 131; 348/552, 584, 589

[56] **References Cited**

U.S. PATENT DOCUMENTS

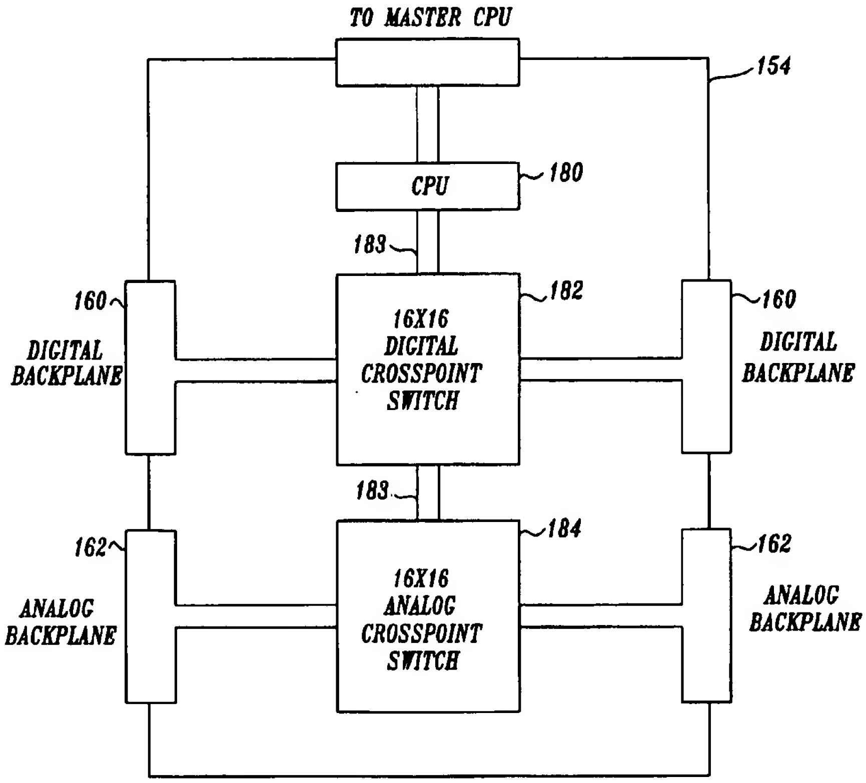
5,721,842	2/1998	Beasley et al.	710/131
5,884,096	3/1999	Beasley et al.	710/131

Primary Examiner—Dung C. Dinh
Attorney, Agent, or Firm—Oblon, Spivak, McClelland, Maier & Neustadt, P.C.

[57] **ABSTRACT**

A computerized switching system for coupling a workstation to a remotely located computer. A signal conditioning unit receives keyboard and mouse signals generated by a workstation and generates a data packet which is transmitted to a central crosspoint switch. The packet is routed through a crosspoint switch to another signal conditioning unit located at a remotely located computer. The second signal conditioning unit applies the keyboard and mouse commands to the keyboard and mouse connectors of the computer as if the keyboard and mouse were directly coupled to the remote computer. Video signals produced by the remote computer are transmitted through the crosspoint switch to the workstation. Horizontal and vertical sync signals are encoded on to the video signals to reduce the number of cables that extend between the workstation and the remote computer. The signal conditioning units connected to the workstations include an onscreen programming circuit that produces menus for the user on a video display of the workstation.

17 Claims, 13 Drawing Sheets



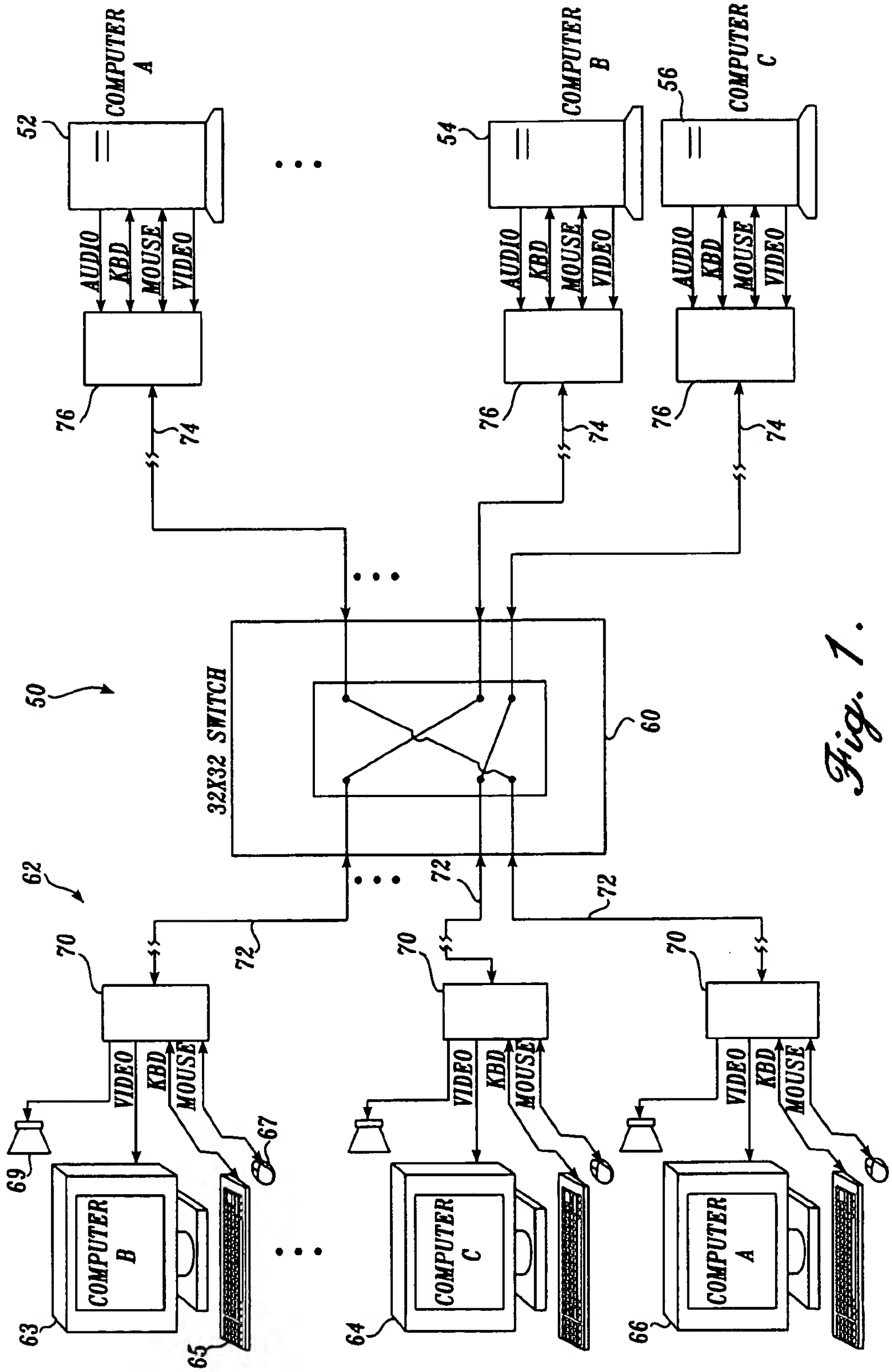

COMPUTER A
52
COMPUTER B
54
COMPUTER C
56
AUDIO
KBD
MOUSE
VIDEO
76
74
50
32X32 SWITCH
60
62
72
70
VIDEO
KBD
MOUSE
67
69
COMPUTER B
63
65
COMPUTER C
64
COMPUTER A
66

Fig. 1.

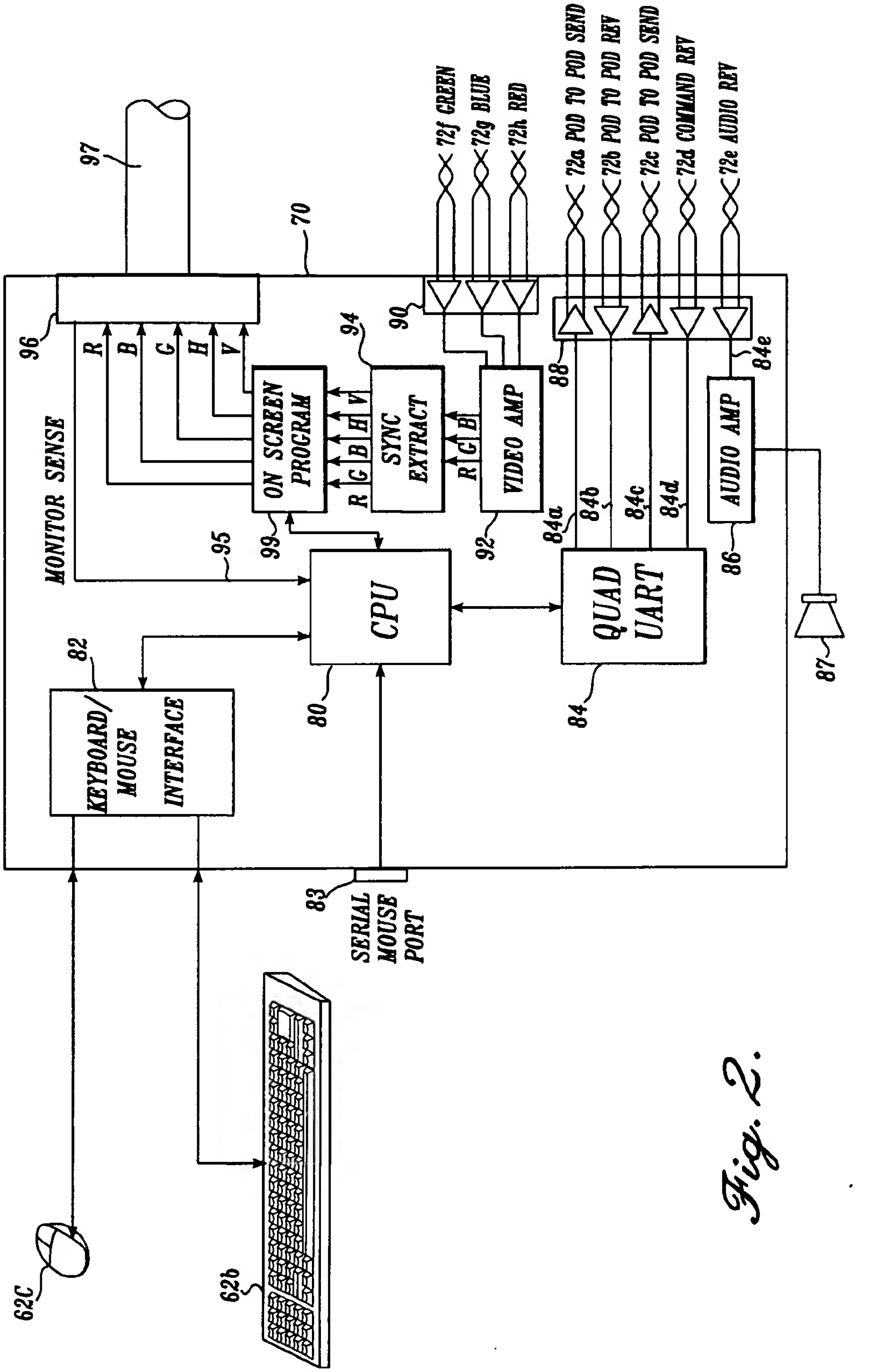

72f GREEN
72g BLUE
72h RED
72a POD TO POD SEND
72b POD TO POD REV
72c POD TO POD SEND
72d COMMAND REV
72e AUDIO REV
97
70
96
90
88
94
R
B
G
H
V
ON SCREEN PROGRAM
R G B H V
SYNC EXTRACT
R G B
VIDEO AMP
AUDIO AMP
84e
84a
84b
84c
84d
MONITOR SENSE
95
99
92
86
CPU
QUAD UART
82
80
84
87
KEYBOARD/ MOUSE INTERFACE
83
SERIAL MOUSE PORT
62C
62b

Fig. 2.

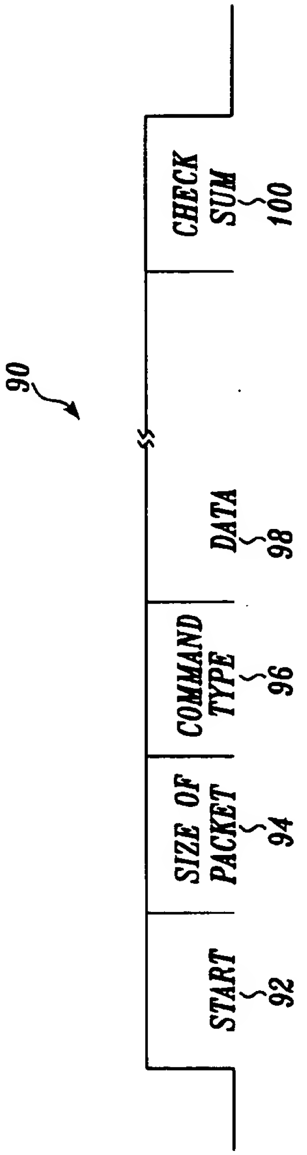

90
START
92
SIZE OF PACKET
94
COMMAND TYPE
96
DATA
98
CHECK SUM
100

Fig. 2A.

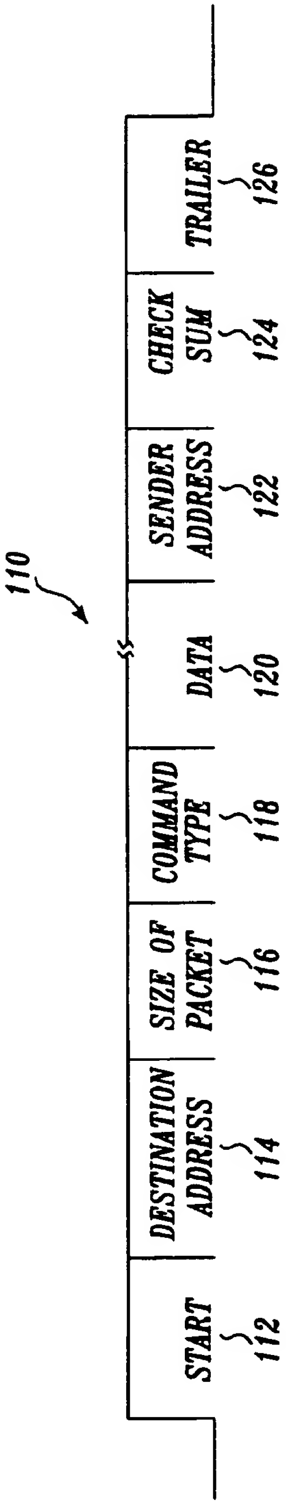

110
START
112
DESTINATION ADDRESS
114
SIZE OF PACKET
116
COMMAND TYPE
118
DATA
120
SENDER ADDRESS
122
CHECK SUM
124
TRAILER
126

Fig. 2B.

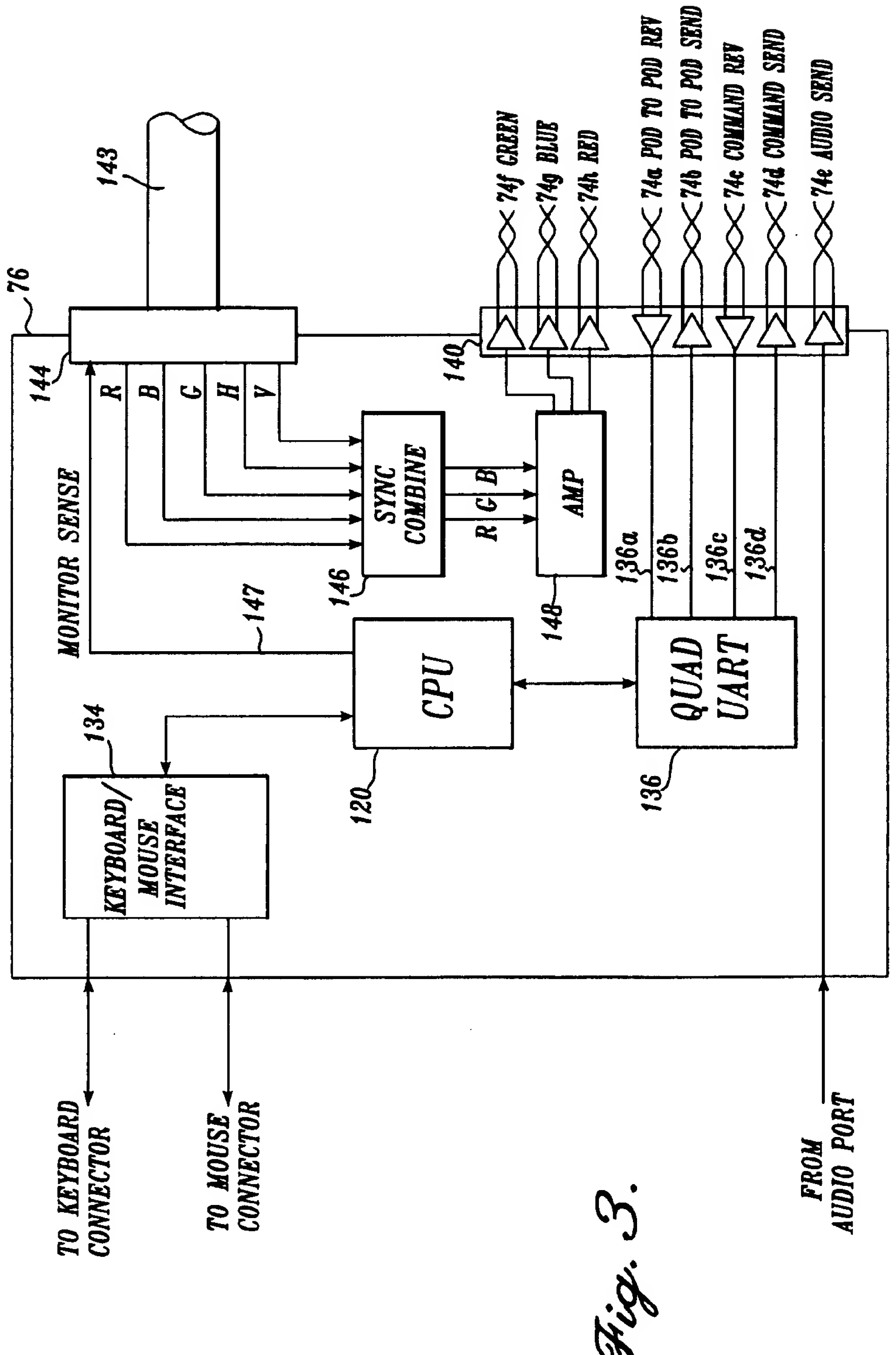

76
143
144
R
B
G
H
V
140
74f GREEN
74g BLUE
74h RED
74a POD TO POD REV
74b POD TO POD SEND
74c COMMAND REV
74d COMMAND SEND
74e AUDIO SEND
MONITOR SENSE
SYNC COMBINE
146
R G B
AMP
148
136a
136b
136c
136d
147
CPU
120
QUAD UART
136
134
KEYBOARD/ MOUSE INTERFACE
TO KEYBOARD CONNECTOR
TO MOUSE CONNECTOR
FROM AUDIO PORT

Fig. 3.

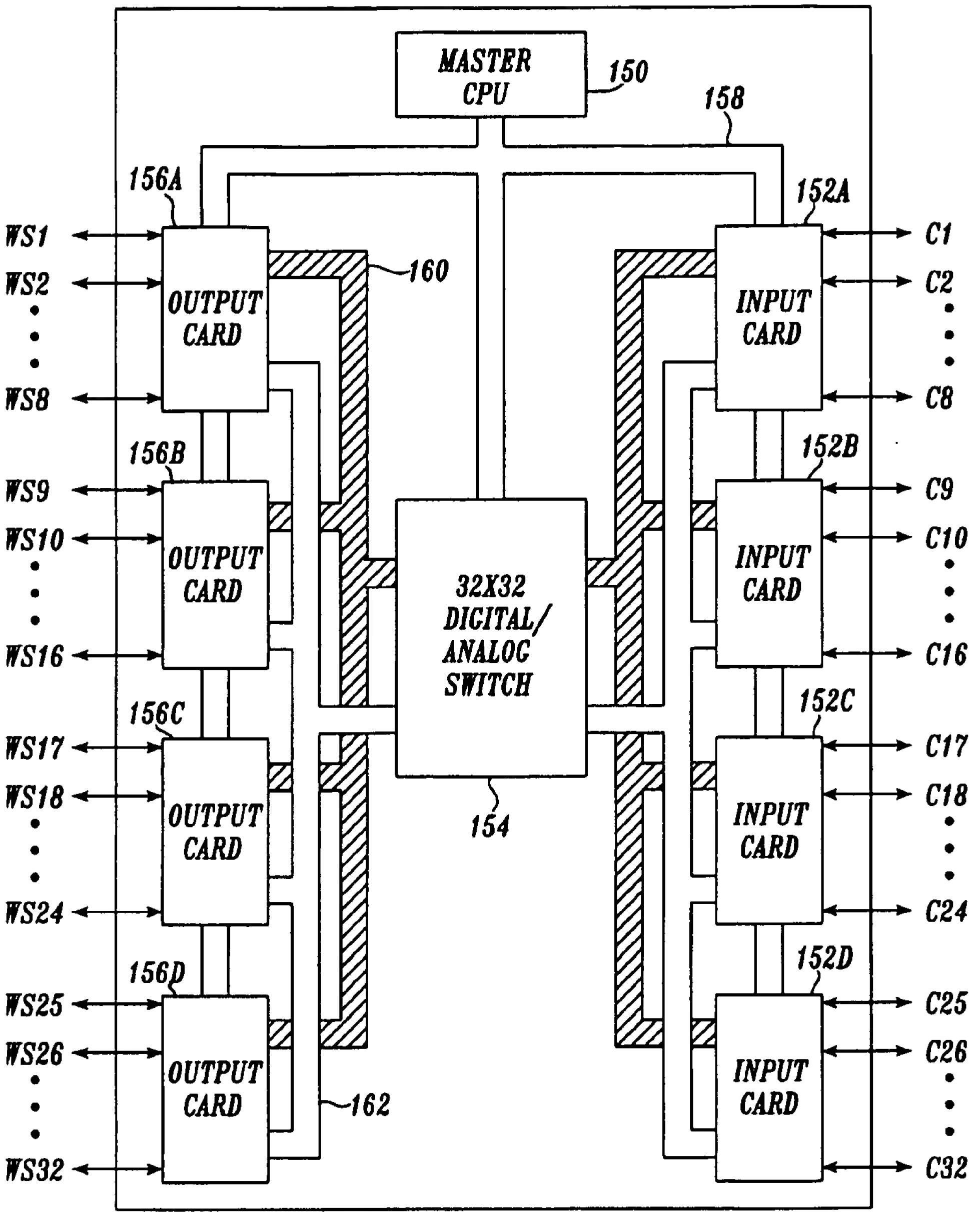

MASTER CPU
150
158
156A
152A
WS1
WS2
WS8
OUTPUT CARD
160
INPUT CARD
C1
C2
C8
156B
152B
WS9
WS10
WS16
OUTPUT CARD
32X32 DIGITAL/ ANALOG SWITCH
INPUT CARD
C9
C10
C16
156C
152C
WS17
WS18
WS24
OUTPUT CARD
154
INPUT CARD
C17
C18
C24
156D
152D
WS25
WS26
WS32
OUTPUT CARD
162
INPUT CARD
C25
C26
C32

Fig. 4.

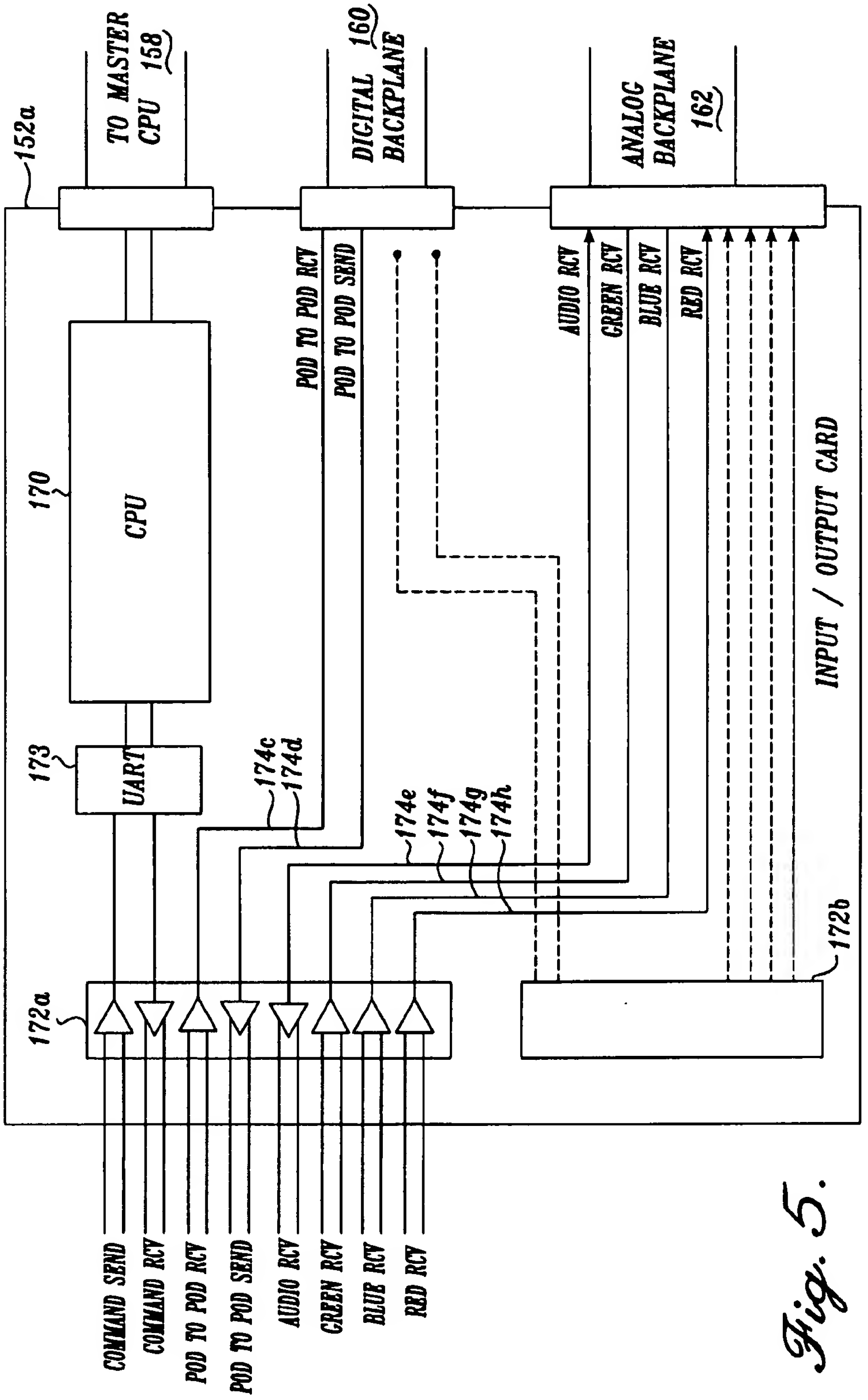

152a
TO MASTER CPU 158
DIGITAL BACKPLANE 160
ANALOG BACKPLANE 162
POD TO POD RCV
POD TO POD SEND
AUDIO RCV
GREEN RCV
BLUE RCV
RED RCV
INPUT / OUTPUT CARD
170
CPU
173
UART
174c
174d
174e
174f
174g
174h
172b
172a
COMMAND SEND
COMMAND RCV
POD TO POD RCV
POD TO POD SEND
AUDIO RCV
GREEN RCV
BLUE RCV
RED RCV

Fig. 5.

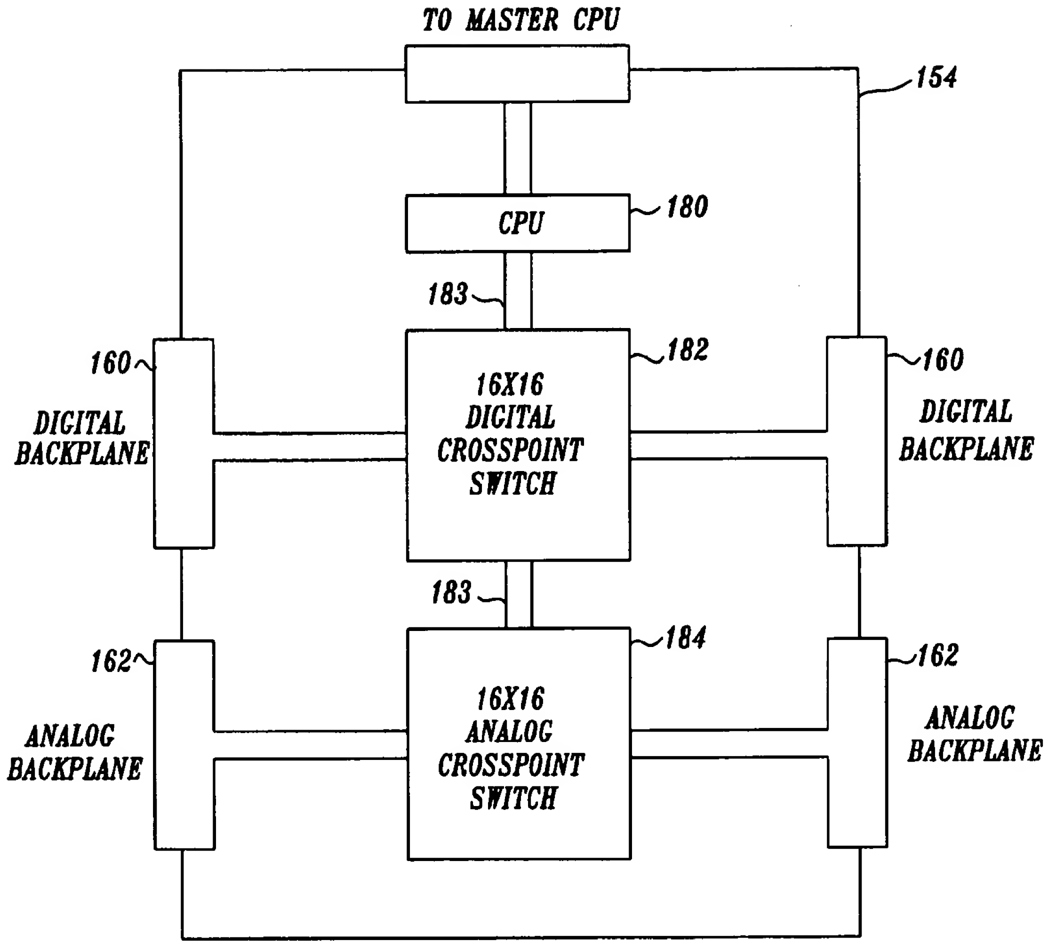

TO MASTER CPU
154
CPU
180
183
182
16X16 DIGITAL CROSSPOINT SWITCH
160
DIGITAL BACKPLANE
160
DIGITAL BACKPLANE
183
184
16X16 ANALOG CROSSPOINT SWITCH
162
ANALOG BACKPLANE
162
ANALOG BACKPLANE

Fig. 6.

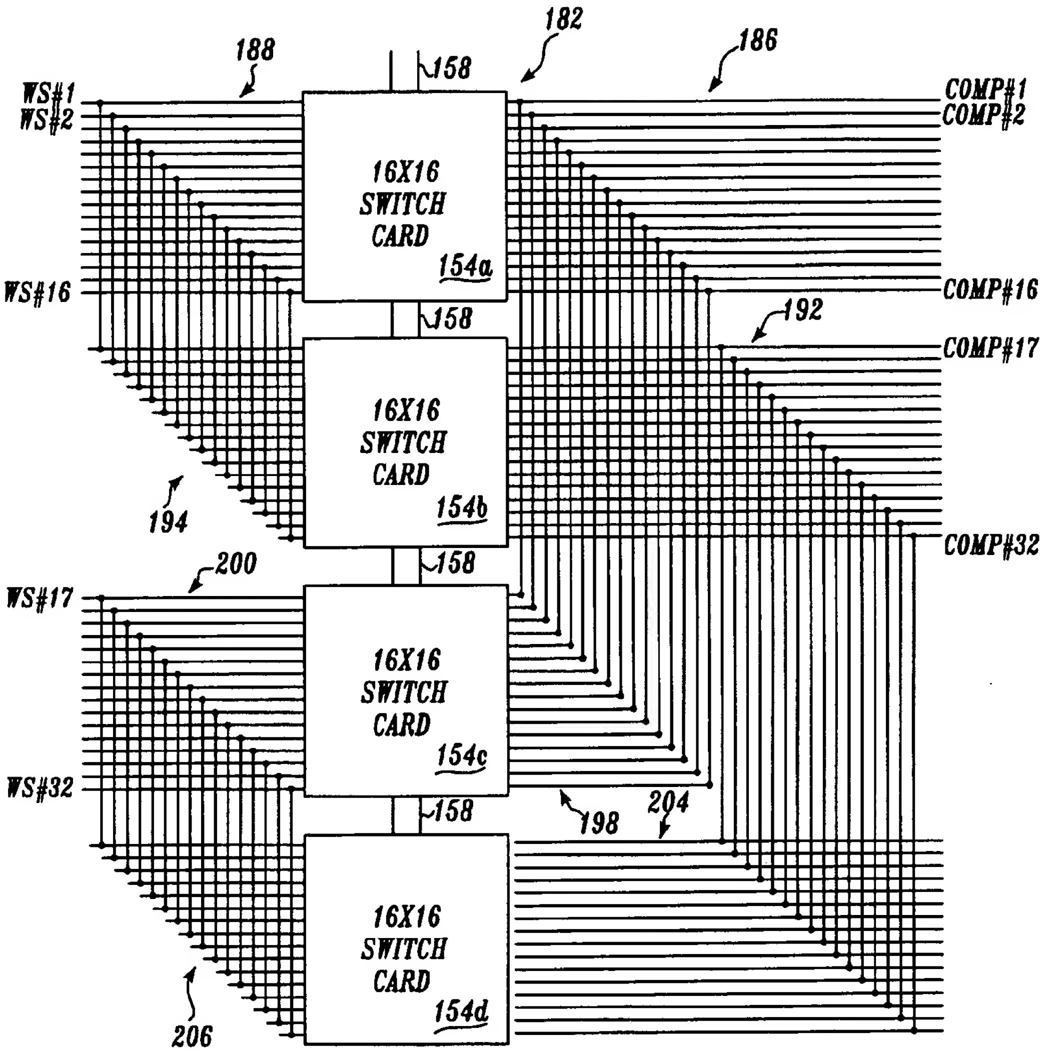

182
188
186
158
WS#1
WS#2
COMP#1
COMP#2
16X16
SWITCH
CARD
154a
WS#16
COMP#16
158
192
COMP#17
16X16
SWITCH
CARD
154b
194
COMP#32
158
200
WS#17
16X16
SWITCH
CARD
154c
WS#32
198
204
158
16X16
SWITCH
CARD
154d
206

Fig. 7.

Fig. 8.

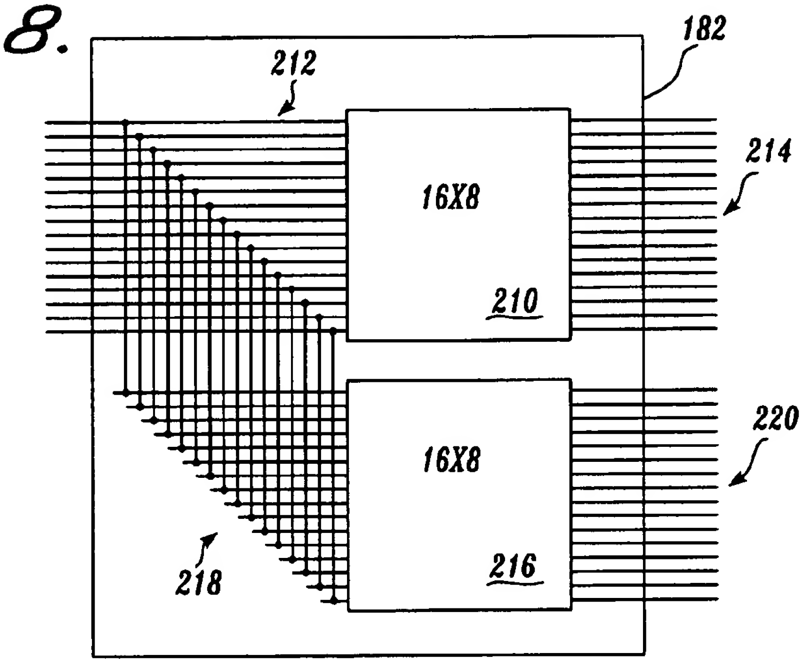
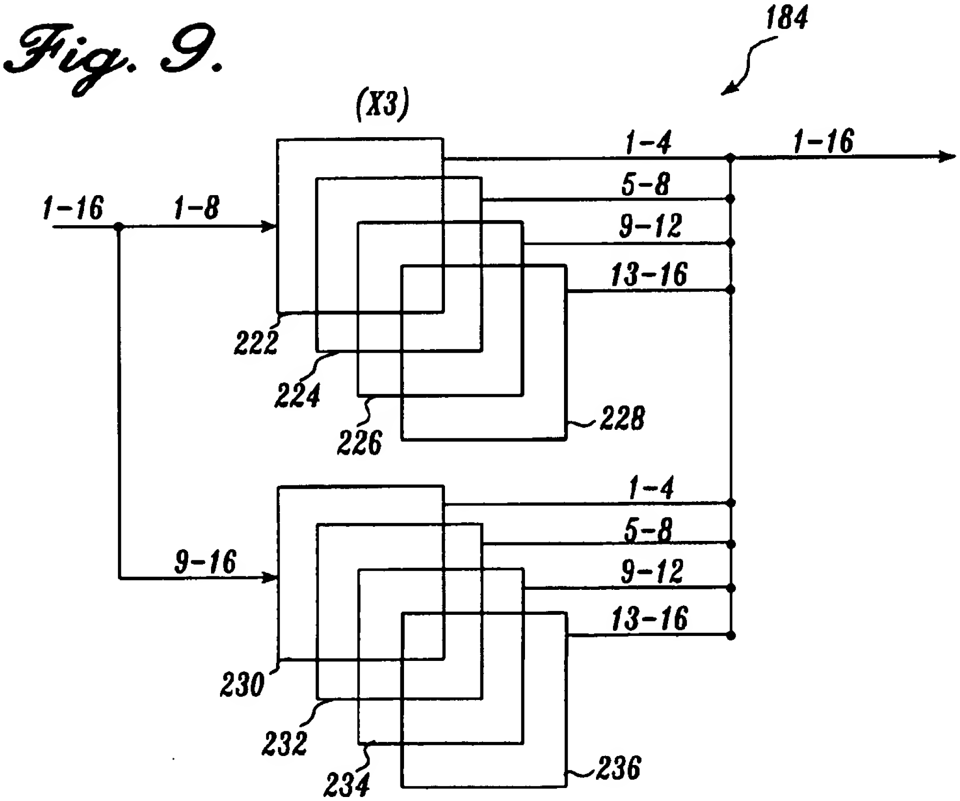


Fig. 9.



THE VIDEO BUS USES 8 8X4 SWITCHES PER COLOR.
TOTAL VIDEO SWITCHES EQUALS 24.

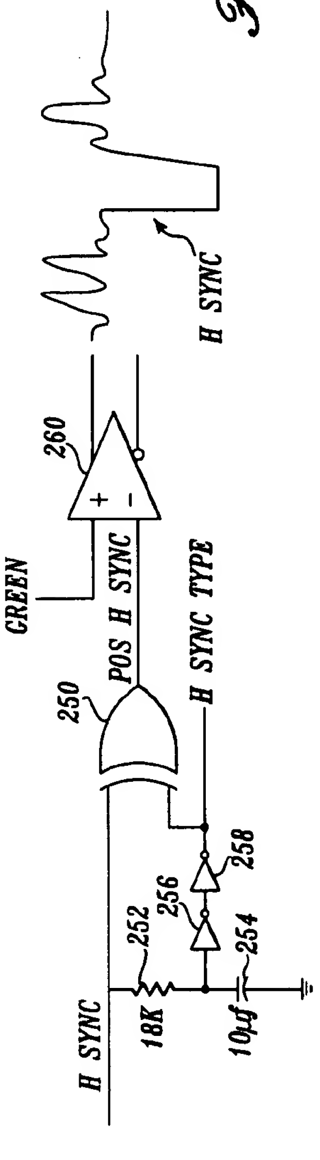


Fig. 10A.

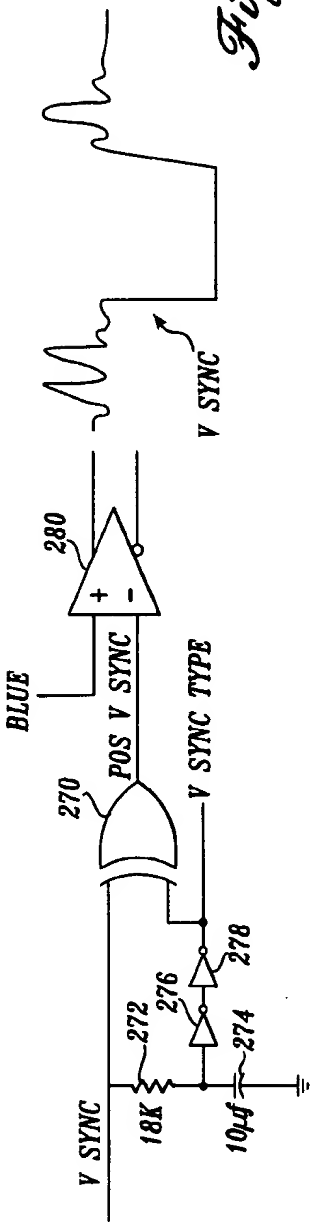


Fig. 10B.

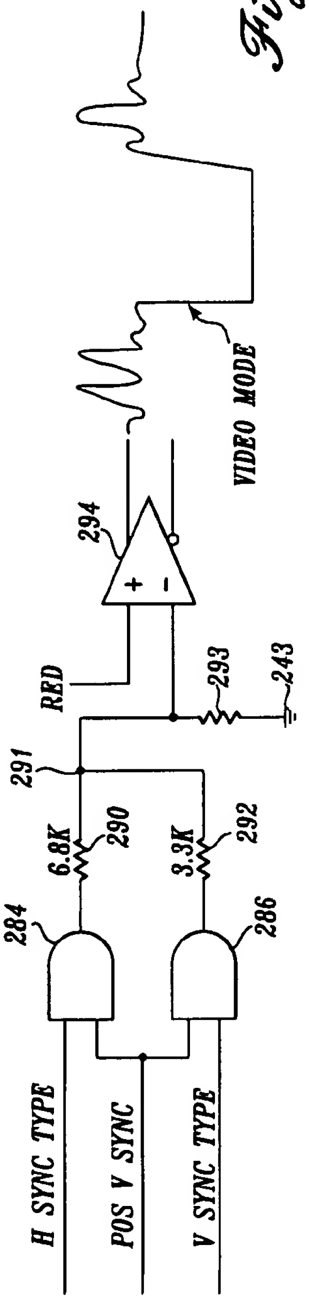


Fig. 10C.

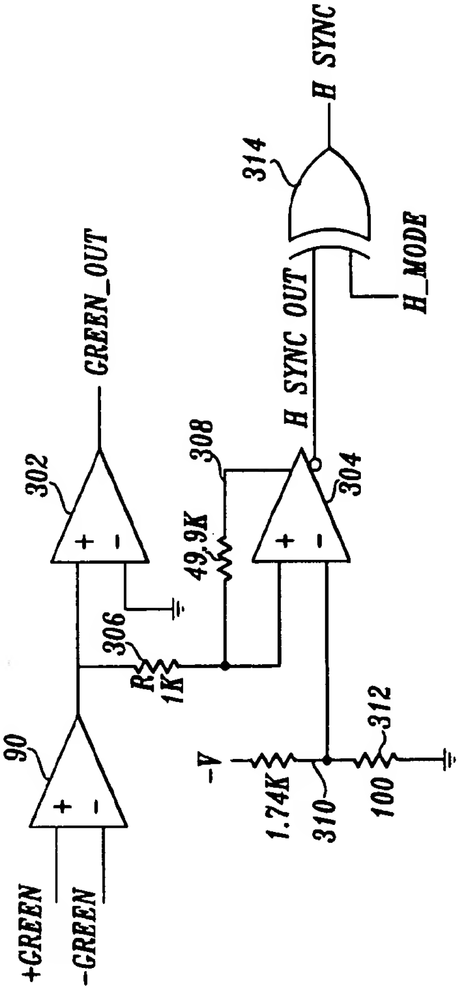


Fig. 11A.

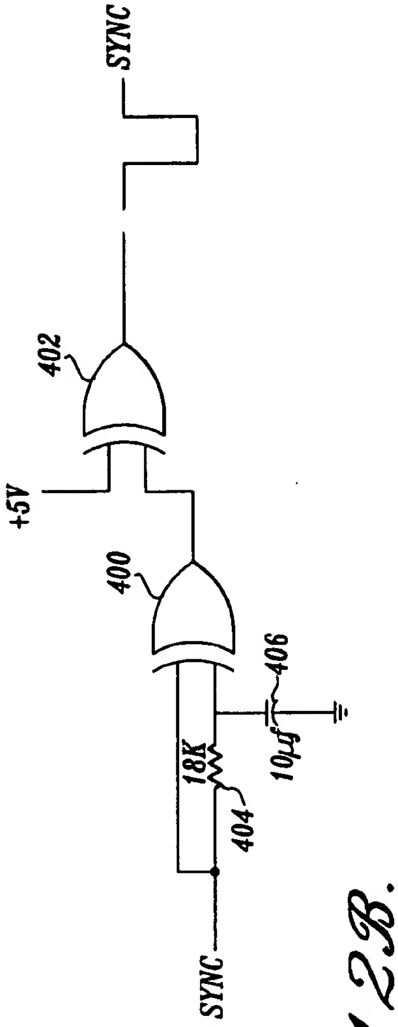


Fig. 12B.

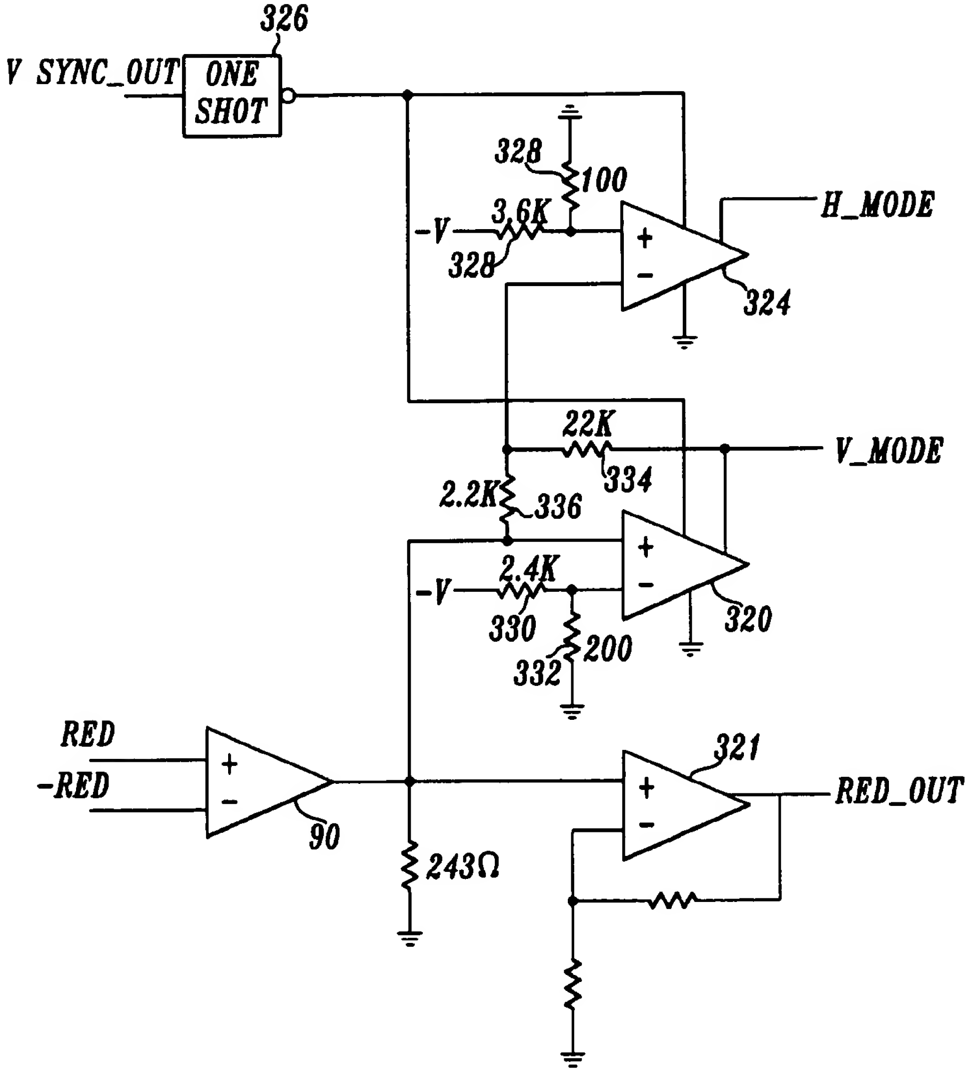

326
V SYNC_OUT
ONE SHOT
328
100
-V
3.6K
328
H_MODE
324
22K
V_MODE
334
2.2K
336
2.4K
-V
330
332
200
320
RED
-RED
90
243Ω
321
RED_OUT

Fig. 11B.

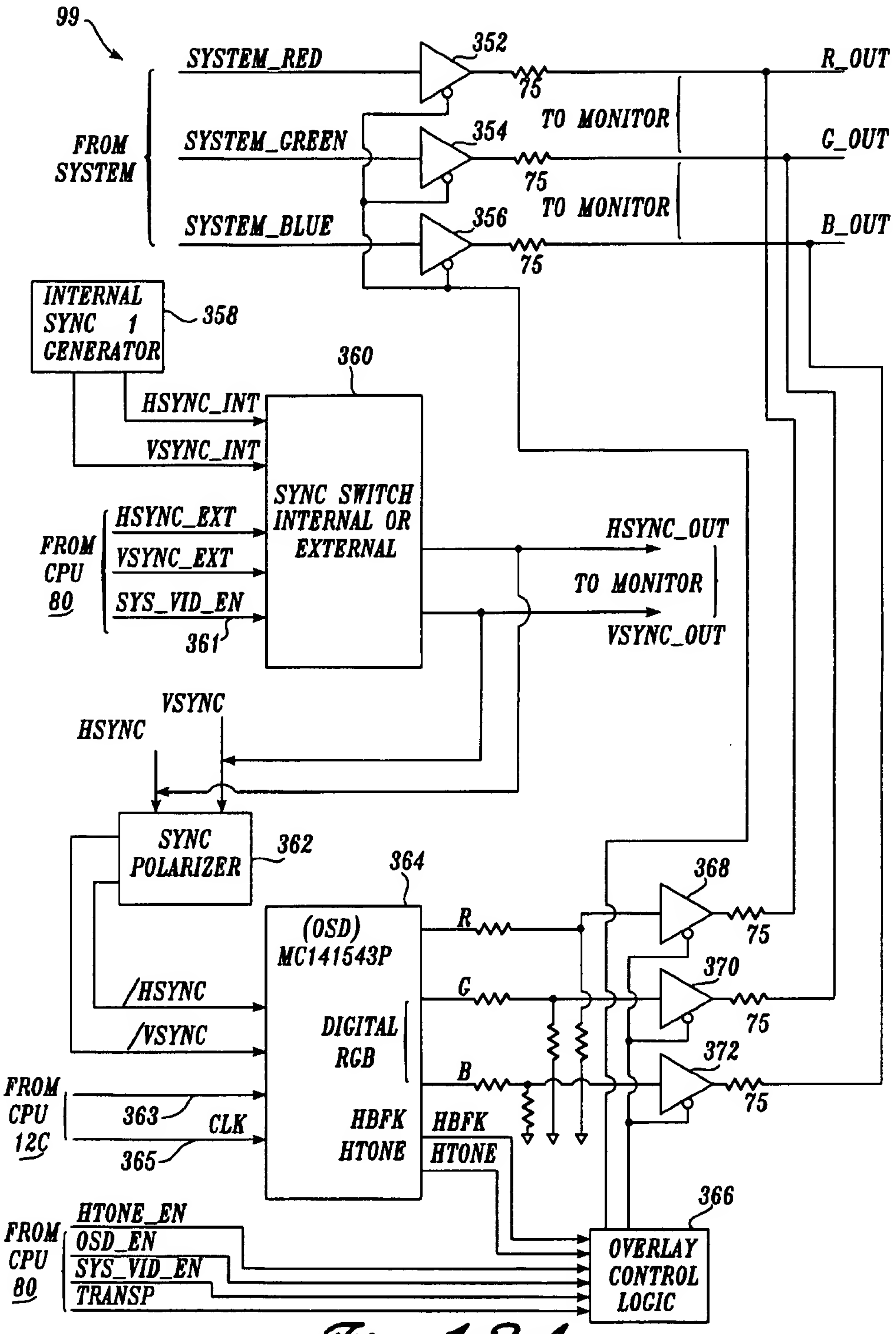

99
SYSTEM_RED
352
75
R_OUT
FROM SYSTEM
SYSTEM_GREEN
354
TO MONITOR
G_OUT
75
SYSTEM_BLUE
356
TO MONITOR
B_OUT
75
INTERNAL SYNC 1 GENERATOR
358
360
HSYNC_INT
VSYNC_INT
SYNC SWITCH INTERNAL OR EXTERNAL
FROM CPU 80
HSYNC_EXT
VSYNC_EXT
SYS_VID_EN
361
HSYNC_OUT
TO MONITOR
VSYNC_OUT
VSYNC
HSYNC
SYNC POLARIZER
362
364
(OSD) MC141543P
368
R
75
/HSYNC
G
370
75
/VSYNC
DIGITAL RGB
B
372
75
FROM CPU 12C
363
CLK
365
HBFK
HTONE
HBFK
HTONE
366
FROM CPU 80
HTONE_EN
OSD_EN
SYS_VID_EN
TRANSP
OVERLAY CONTROL LOGIC

Fig. 12A.

COMPUTER INTERCONNECTION SYSTEM HAVING ANALOG OVERLAY FOR REMOTE CONTROL OF THE INTERCONNECTION SWITCH

RELATED APPLICATIONS

The present application is a continuation of U.S. patent application Ser. No. 08/969,723, filed Nov. 12, 1997 now U.S. Pat. No. 5,884,096, which is a continuation of U.S. patent application Ser. No. 08/519,193, filed Aug. 25, 1995, now issued as U.S. Pat. No. 5,721,842, the benefit of which is claimed under 35 U.S.C. §120.

FIELD OF THE INVENTION

The present invention relates to systems for interconnecting remotely located computers.

BACKGROUND OF THE INVENTION

In a typical local computer network there are a number of client computers that are coupled via a communication link to a number of network server resources. These resources include file servers, print servers, modem servers, and CD-ROM servers for example. Each server is usually a stand alone computer with its own keyboard, mouse and video monitor. Each client computer can utilize the functions provided by the server computers through the communication link.

Most computer networks have one or more system administrators, i.e. human operators, for the server computers. The system administrators monitor the operation of the software running on the server computers, load new software packages, delete outdated files and perform other tasks necessary to maintain the operation of the network. While most administrator tasks (modifying software, deleting files, etc.) can be performed over the network from a client computer, there are some situations where the network administrators must be physically located at the server computers for direct access to and operation of them. For example, it is not possible to reboot a server computer over the network. If the server computers are not close together, the time required for a task as simple as rebooting can be substantial.

Although it is possible to run dedicated communication links to each server computer in order to allow a system administrator to operate the network from a central location, a large number of cables are required for anything other than a very simple network.

SUMMARY OF THE INVENTION

The present invention provides a computerized switching system that allows centrally located network administrators to operate multiple server computers over long distances without requiring a complicated wiring scheme. In general, the switching system allows data transmission between a workstation and a remotely located server computer. A signal conditioning unit receives keyboard and mouse signals from a workstation and generates a serial data packet which is transmitted to a central crosspoint switch. The crosspoint switch routes the keyboard/mouse packet to another signal conditioning unit that is coupled to the remotely located server computer. The signal conditioning unit coupled to the server computer decodes the keyboard/mouse packet and applies the signals to a keyboard and mouse connector on the remote computer in the same manner as if the mouse and keyboard were directly coupled to the remote computer.

Video signals produced by the remote computer are transmitted through the crosspoint switch to the workstation. In order to minimize the number of wires extending between the remote computer and the workstation, the horizontal and vertical sync signals as well as a mode signal are encoded with the analog video signals. The present embodiment of the invention allows any of thirty-two workstations to be connected to any of thirty-two remotely located server computers.

BRIEF DESCRIPTION OF THE DRAWINGS

The foregoing aspects and many of the attendant advantages of this invention will become more readily appreciated as the same becomes better understood by reference to the following detailed description, when taken in conjunction with the accompanying drawings, wherein:

FIG. 1 is a pictorial diagram of a computerized switching system, according to the present invention, a number of workstations and a number of remotely-located computers;

FIG. 2 is a block diagram of a signal conditioning unit (pod) that is coupled to a workstation;

FIG. 2A is a timing diagram of a serial pod to pod packet that is transmitted by the signal conditioning unit shown in FIG. 2;

FIG. 2B is a timing diagram of a data packet that is routed within the central crosspoint switch;

FIG. 3 is a block diagram of a signal conditioning unit (pod) that is coupled to a remote computer system;

FIG. 4 is a block diagram of a crosspoint switch according to the present invention that routes data between a workstation and a remote server computer;

FIG. 5 is a block diagram of an input/output card that is utilized to send and receive signals at the crosspoint switch;

FIG. 6 is a block diagram of a switch card that routes signals through the crosspoint switch;

FIG. 7 is a schematic diagram showing the interconnection of four switch cards to create a 32x32 switch utilized in the crosspoint switch of the present invention;

FIGS. 8 and 9 are schematic diagrams showing how a digital and analog 16x16 switch is constructed;

FIGS. 10A–10C are schematic diagrams of circuits for encoding horizontal sync, vertical sync and video mode signals onto an analog video signal according to another aspect of the present invention;

FIGS. 11A and 11B are schematic diagrams of circuits for extracting the encoded horizontal and vertical sync signals and the mode signal from an analog video signal;

FIG. 12A is a circuit diagram of an onscreen programming circuit that produces video displays on the workstation's monitor according to yet another aspect of the present invention; and

FIG. 12B is a circuit diagram of a circuit that inverts the polarity of horizontal and vertical sync signals that is used within the onscreen programming circuit of FIG. 12A

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

The present invention is a computerized switching system for allowing a number of computer workstations to be coupled to a number of remotely-located server computers. In the presently preferred embodiment of the invention, up to thirty-two workstations can be connected to any of thirty-two remote computer systems. However, those skilled in the art will recognize that the number of possible interconnections can easily be modified for the environment in which the invention is to be used.

Referring now to FIG. **1**, the computerized switching system or crosspoint switch according to the present invention allows a number of server computers **52, 54, 56** to be coupled to a number of workstations **62, 64, 66**. Each workstation includes a video monitor **63**, a keyboard **65** and a cursor control device such as a mouse **67**. In accordance with the present invention, signals from the keyboard **65** and the mouse **67** are received by a signal conditioning circuit or pod **70**. The pod transmits the keyboard and mouse signals over a communication link **72** to a central crosspoint switch **60**. After being routed through the crosspoint switch **60**, the keyboard and mouse signals are retransmitted on another communication link **74** to a pod **76**, which is coupled to the remotely-located server computer. The pod **76** supplies the keyboard and mouse signals through appropriate connectors to keyboard and mouse input ports of the remote computer, just as if the keyboard **65** and mouse **67** were directly coupled to the keyboard and mouse input ports.

Audio and video signals produced by the remote server computer **52, 54** or **56** are received by the associated pod **76** and transmitted in the reverse direction along the communication link **74** to the central crosspoint switch **60**. The central crosspoint switch routes the audio and video signals to one of the communication links **72** for transmission to a pod **70**. The pod **70** then supplies the audio and video signals to the associated video monitor **63** and a speaker **69** of the workstation. From a user's perspective, the work station appears as if it is directly coupled to the remote server computer.

FIG. **2** is a block diagram of a pod **70**. As described above, the pod operates to receive the mouse and keyboard signals and to transmit them through the crosspoint switch to a remotely-located server computer system. In addition, the pod receives video and audio signals from the remote server computer by way of the central crosspoint switch and supplies them to the video monitor and speaker of the workstation.

The pod **70** generally comprises a central processing unit (CPU) **80** having its own random access and read only memories. A keyboard/mouse interface **82** is coupled to the CPU **80** to receive and condition the electronic signals from the keyboard **65** and mouse **67**. As the user moves the mouse or types on the keyboard, the keyboard/mouse interface **82** generates an interrupt signal that is fed to the CPU **80**. The CPU **80** then reads the digitally buffered keyboard and mouse signals from the keyboard/mouse interface **82** and converts the signals into a data packet that is transmitted to the remote computer.

As shown in FIG. **2A**, the pod to pod data packet **90** begins with a unique character **92** that marks the beginning of the data packet followed by a byte **94** that indicates the length of the packet. The next byte **96** identifies the type of data (mouse, keyboard, monitor type etc.) that the packet represents. The next series of bytes **98** represents the keyboard/mouse data to be transmitted to the server computer. Finally, a checksum byte **100** allows for the correction of errors that may occur during transmission.

It should be noted that the pod to pod packets are not limited to carrying keyboard and mouse data. The packets allow the pod at the work station to "talk to" the pod at the remote computers. Each pod acknowledges to the other that a packet was received correctly and in case of an error requests that a packet be retransmitted.

After the CPU **80** has assembled the pod to pod packet, the packet is transmitted to a quad UART **84**, which transmits and receives serial data on four leads **84*a*–84*d***. The pod to pod packet is serialized and transmitted on the lead **84*a*** to a differential line driver/receiver **88** that transmits and receives data on a number of twisted-pair cables **72*a*–72*e***, that are coupled to the central crosspoint switch **60** (shown in FIG. **1**). In the presently preferred embodiment of the invention, the differential line drivers/receivers are model Nos. DS8921, manufactured by National Semiconductor. The drivers transmit a positive version of the data on one wire of the twisted-pair cable and the inverse of the data on the other wire of the twisted pair. This allows the data to be transmitted along cables up to 500 feet in length without the use of additional amplifiers.

As the user is operating the remote server computer, the remote computer may transmit commands which affect the operation of the mouse and keyboard. These include the mouse sensitivity, the keyboard repeat rate, activating one or more LEDs on the keyboard (such as the number lock, capital letter lock, etc.). The keyboard/mouse commands contained in a pod to pod packet transmitted from the remote computer are received on twisted-pair cable **72*b*** by the differential line driver/receiver **88**. The UART **84** converts the received serial keyboard/mouse commands into a parallel format and supplies the data to the CPU **80**. The CPU **80** then generates the appropriate signals which are fed to the keyboard/mouse interface **82** and applied to the keyboard **62*b*** and mouse **62*c***.

Video signals transmitted from the remote server computer are received on three sets of twisted-pair cables **72*f*, 72*g***, and **72*h*** by a set of differential line receivers **90**. The output signals produced by the differential line receivers **90** are supplied to a video amplifier **92**. The output of the video amplifier is coupled to a sync extract circuit **94** which removes an embedded horizontal and vertical sync signal as well as a mode signal from the green, blue and red video signals respectively. The sync extract circuit **94** supplies the red, blue, and green analog video signals as well as the horizontal and vertical sync signals on separate leads to an onscreen programming circuit **99** that is described in further detail below. The onscreen programming circuit **99** feeds the video signals to a connector **96**, which is coupled to the video monitor of the workstation by a conventional video cable **97**. As will be described in further detail below, the horizontal and vertical sync signals are embedded into the green and blue color video signals in order to minimize the number of wires that extend between the workstation and the remote server computer as well as to reduce the complexity of the crosspoint switch.

The CPU **80** also reads a set of four monitor sense leads **95** to determine what type of monitor is connected to it. Monitor sense data is generated and transmitted in a pod to pod packet as shown in FIG. **2A**. The remote computer receives the monitor data and supplies it to the remote computer in order to adjust its video signals accordingly.

In addition to transmitting and receiving keyboard and mouse signals from the remote computer, the pod **70** can communicate with the central crosspoint switch. Data to be transmitted to the central crosspoint switch are sent on a twisted pair cable **72*c*** while data transmitted from the central crosspoint switch are received on a twisted pair cable **72*d***.

Commands sent between the pod **70** and the central crosspoint switch allow a user to connect the work station to another remote computer, allow the central crosspoint switch to interrogate the status of the pod, update the firmware of the pod, etc. using the packet structure shown in

FIG. 2B as will be described below. When the user wishes to send a command to the central crosspoint switch, a special sequence of keystrokes is used. In the present embodiment of the invention, all commands are preceded by the "printscreen" key and end with the "enter" key. The CPU **80** parses the keyboard strokes for these keys and analyzes the keystrokes to determine the destination of the command. If the command is directed to the pod itself, no data packet is produced. If the command is directed to the remote computer, a pod to pod packet is generated and transmitted. If the command is directed to the central crosspoint switch, the CPU assembles a command packet that is transmitted to the central crosspoint switch on the twisted pair cable **72***c*.

A block diagram of a pod **76** that is coupled to the remote server computers is shown in FIG. **3**. The pod **76** includes a central processing unit (CPU) **120** that is coupled to a keyboard/mouse interface **134**. The keyboard/mouse interface **134** supplies signals to and receives signals from the server computer's keyboard and mouse connectors. The keyboard and mouse signals from the computer's keyboard and mouse connectors are read by the CPU **120** and assembled into a pod to pod packet in the same manner as the pod to pod packet described above and shown in FIG. 2A. The pod to pod packet produced by the CPU **120** is delivered to a QUAD UART **136** that transmits the packet serially over a lead **136***b* to a differential line driver **140**. The differential line driver drives a twisted-pair cable **74***a* that is coupled to the central crosspoint switch.

A pod to pod packet that is transmitted from a workstation is received on a twisted-pair cable **74***b* and supplied to differential line receiver **140**. The output signal of the differential line receiver is supplied to the QUAD UART **136** which converts the packet from a serial format to a parallel format. The CPU reads the packet and then transmits the received keyboard and mouse signals to the keyboard and mouse interface **134** where the signals are supplied to the remote computer's keyboard and mouse connectors in the same manner as if the keyboard and mouse were directly connected to the remote server computer. The particular format of the signals applied to the keyboard and mouse connectors may vary with the type of the remote computer. The CPU within the pod **76** is therefore programmed to translate the signals into their proper format.

Commands sent from the pod **76** to the central crosspoint switch allow the remote computer to interrogate the status of the pod, update the firmware of the pod etc. using the packet structure of FIG. **2B**. As with the user pod, all commands are preceded with the "printscreen" key and end with the "enter" key. The CPU **120** parses the keyboard strokes for these keys and analyzes the keystrokes to determine the destination of the command. If the command is directed to the pod **76**, no data packet is produced. If the command is directed to the workstation, a pod to pod packet is generated and transmitted. If the command is directed to the central crosspoint switch, the CPU assembles a command packet that is transmitted to the central crosspoint switch on a twisted pair cable **74***d*.

The signals from the remote computer's video port are supplied through a video cable **143** to a connector **144**. As will be described below, the red, green and blue analog video signals along with the horizontal and vertical sync signals are supplied to a sync combine circuit **146** that encodes the horizontal and vertical sync signals onto the green and blue analog video signals respectively. The current mode of the monitor (i.e., the correct polarity of the horizontal and vertical sync pulses) is encoded by the sync combine circuit **146** onto the red analog video signal. The output of the sync combine is supplied to an amplifier **148** that conditions the signals and supplies the video signal to three differential line drivers **140** that transmit the signals over three separate twisted-pair cables **74***f*, **74***g*, and **74***h* to the central crosspoint switch.

The monitor sense data received from a remote workstation is decoded by the CPU **120** and supplied to a set of monitor sense leads **147**. The remote computer receives the monitor sense data on these leads and adjusts its video signals for the particular monitor that is displaying the video signals.

The audio signals produced by the remote computer are supplied to a differential line driver **140** and are transmitted over a twisted-pair cable **74***c* to the central crosspoint switch.

FIG. **4** is a block diagram of the central crosspoint switch. The central switch **60** includes a master central processing unit (CPU) **150**, a number of input cards **152**, a number of switch cards **154** and a number of output cards **156**. Each of the input cards transmits signals to and receives signals from up to eight of the remotely located server computers while each of the output cards transmits to and receives signals from up to eight of the remotely located workstations. The master CPU **150** is coupled to each of the input cards **152**, the switch cards **154** and each of the output cards **156** by a digital bus **158**. Together the master CPU, input cards, switch cards and output cards are connected via a local area network.

Pod to pod packets are routed from an input card through the switch card to an output card and vice versa on a digital backplane **160**. The analog video and audio signals are transmitted between the input cards, the switch card **154** and the output cards **156** on a separate analog backplane **162**.

A block diagram of an input card **152** is shown in FIG. **5**. The output cards **156** are identical to the input cards except that the direction of the audio/video signals is reversed and therefore will not be discussed separately. The input card **152** includes its own CPU **170** that transmits and receives data from the master CPU **150**. Signals transmitted from the remote server computer are received by a set of differential line drivers/receivers **172***a–b*. Commands sent from the remote computer to the central crosspoint switch are received by an octal UART **173** where the commands are converted from a serial to a parallel format. The UART feeds the commands to the CPU **170** where they are interpreted and forwarded to the master CPU **150**.

To transmit data between the input, output and switch cards of the crosspoint switch, the data is packetized in the format shown in FIG. **2B** by the CPU of the card sending the packet. A packet begins with a unique character **112** that marks the beginning of the packet. A destination address **114** follows the start character. The address uniquely identifies one of the cards in the crosspoint switch. A byte **116** indicates the size of the packet while a byte **118** indicates the type of data included in the packet. A series of bytes **120** are the data to be transmitted from one card to another. Following the data, a byte **122** indicates the sending card's unique address. A checksum byte **124** follows the sender's address and a unique character **126** is sent as a trailer. The transmission of all data packets between the cards of the crosspoint switch is controlled by the master CPU **150**.

Returning to FIG. **5**, commands generated by the CPU **170** to be transmitted to the pod that is coupled to a remote server computer are transmitted on a lead **174***b* to a differential line driver **172**. Pod to pod packets received from the central computer are routed through the input card on a lead

174*c* to the digital backplane **160**. Similarly, pod to pod packets transmitted from the remote workstation are received from the digital backplane, routed through the input card on a lead **174***d* and supplied to the differential line driver **172***a*.

In order to shield the video signals from the noise on the digital backplane, the video and audio signals transmitted from the remotely located server computer are routed on a separate analog backplane **162**. The audio signals received from the remote computer are routed through the input card on a lead **174***e* and applied to the analog backplane **162**. Video signals are received by the differential line receivers **172***a* and routed through the input card on leads **174***f–h* to the analog backplane.

In the present embodiment of the invention, each input card includes up to eight sets of differential line drivers/receivers **172***a*–**172***f* (the remaining six driver/receivers not shown) to receive signals from up to eight remotely located server computers. The signals from each remotely located computer are routed through the input card to the digital and analog backplanes in the manner described above.

FIG. **6** is a block diagram of a switch card **154**. The switch card includes its own central processing unit (CPU) **180**. The CPU **180** transmits and receives signals from the master CPU **150** in order to control the position of a 16x16 digital crosspoint switch **182** and a 16x16 analog crosspoint switch **184** using a set of control leads **183**. The digital crosspoint switch **182** connects the keyboard/mouse signals transmitted between a workstation and a remote server computer as well as audio signals generated by the remote server computer to the workstation. The analog crosspoint switch **184** transmits the video signals between a remote server computer and any of the workstations.

FIG. **7** shows how the digital backplane portion of the 32x32 crosspoint switch is configured using four switch cards **154***a*, **154***b*, **154***c* and **154***d* in order to transmit signals between **32** workstations and **32** remotely located server computers. The switch card **154***a* has sixteen input lines **186** that are coupled to sixteen remotely located server computers and sixteen output lines **188** that are coupled to sixteen workstations. The switch card **154***b* has sixteen input lines coupled to another sixteen remotely located server computers and sixteen output lines **194** that are coupled to each of the sixteen output lines **188** of the switch card **154***a*. The switch card **154***c* has sixteen input lines **198** that are coupled to the sixteen input lines **186** of the switch card **154***a*. The sixteen output lines **200** of the switch card **154***c* are coupled to another sixteen remotely located workstations. The switch card **154***d* has sixteen input lines **204** that are coupled to each of the sixteen input lines **192** of the switch card **154***b*. The sixteen output lines **206** of the switch card **154***d* are coupled to the sixteen output lines **200** of the switch card **154***c*. The analog backplane is constructed in a similar fashion as the digital backplane described above. As can be seen, the arrangement of the switch cards **154***a*, **154***b*, **154***c* and **154***d*, allows data from any one of thirty-two remotely located computers to be coupled to any one of thirty-two remotely located workstations.

A switching arrangement of the type shown in FIG. **7** is required for each signal that is to be transmitted between the remotely located server computer to a corresponding workstation. In the present embodiment of the invention, each workstation sends and receives pod to pod packets as well as receives audio and video signals from the remote computer. Therefore, for the 32x32 digital switch shown in FIG. **6**, the digital backplane includes two sets of switches of the type shown in FIG. **7** and the analog backplane includes another four sets of switches for the video and audio signals.

In the presently preferred embodiment of the invention, the digital 16x16 switches **182** are implemented using a pair of 16x8 digital switches as shown in FIG. **8**. Each 16x16 switch comprises switches **210** and **216**. The switch **210** has sixteen input lines **212** and eight output lines **214**. The switch **216** has sixteen input lines **218** that are coupled to each of the input lines **212**, and eight output lines **220**. In the presently preferred embodiment of the invention, each of the 16x8 switches **210** and **216** are part numbers CD22M34945Q, manufactured by Harris.

The analog backplane on which the video signals are transmitted is configured in the same fashion as the switch shown in FIG. **7**. However, because of the greater bandwidth required, each 16x16 switch **184** is implemented using eight 8x4 analog switches model no. DG884DN, manufactured by Siliconix. As can be seen in FIG. **9**, a 16x16 analog switch is implemented using switches **222**, **224**, **226** and **228** each having eight input lines and four output lines. The input lines of switches **222**, **224**, **226** and **228** are connected in parallel. A second set of switches **230**, **232**, **234** and **236**, each having eight input lines and four output lines. The input lines of switches **230**, **232**, **234** and **236** are connected in parallel. The outputs of switch **230** are coupled in parallel with the outputs of switch **222**, and the outputs of switch **232** are coupled in parallel with the outputs of switch **224**. The outputs of switch **234** are coupled in parallel with the outputs of switch **226** and the outputs of switch **236** are coupled in parallel with the outputs of switch **228**.

To minimize the number of wires that must extend from the remote computer to the workstation, the present invention encodes the horizontal and vertical sync signals onto the analog color video signals transmitted from the remote computer. FIGS. **10**A–**10**C show the details of the sync combine circuit **146** (FIG. **3**) that encodes the vertical and horizontal sync signals as well as the mode signal of the monitor. FIG. **10**A shows a circuit that encodes the horizontal sync signal onto the green video signal produced by a remote computer. The circuit includes an exclusive or (XOR) gate **250** having a first input that receives the horizontal sync signal produced by the computer system. A resistor **252** and capacitor **254** are connected in a series between the first input of the XOR gate and ground. At the junction of the resistor **252** and the capacitor **254** are two series connected inverting gates **256** and **258**. The output of the inverter **258** is supplied to a second input of the XOR gate **250**.

The XOR gate **250** operates to encode the horizontal signal as a positively going pulse no matter what the normal state of the horizontal sync signal is. The voltage on the capacitor **254** is equal to the average valve of the horizontal sync signal. The output of the inverting gate **258** has a logic level equal to the non-active state of the horizontal sync signal. The output of the XOR gate **250** is coupled to an inverting input of an amplifier circuit **260**. The non-inverting input of the amplifier **260** is connected to receive the green analog video signal. When the horizontal sync signal is in its normal state, the output of the amplifier **260** follows the green analog video signal. However, when the horizontal sync signal is activated, the active video is at zero volts and the amplifier **260** produces a negative going horizontal sync pulse.

FIG. **10**B shows a circuit that encodes the vertical sync signal onto the blue analog video signal produced by the remote computer. The circuit comprises an exclusive or

(XOR) gate **270**, a resistor **272**, capacitor **274** and a pair of inverters **276, 278** that are connected in the same way as the horizontal sync circuit shown in FIG. **10A** and described above. The output of the XOR gate is always a positive going pulse when the vertical sync signal is activated. The output of the XOR gate is fed to the inverting input of an amplifier **280**. When the vertical signal is in its normal state, the output of the amplifier **280** follows the blue analog video signal. However, when the vertical sync signal is activated, a negative going pulse, V-sync, is created by the amplifier.

FIG. **10C** is an electronic circuit that encodes the mode of the video monitor. The mode refers to the polarity of the horizontal and vertical sync signals. Changes in the mode affect the size of the video display produced by a video monitor. To encode the mode of the video signal, the circuit shown in FIG. **10C** is used. The circuit comprises two AND gates **284** and **286**. The AND gate **284** has one input coupled to the output of the inverter **258** (shown in FIG. **10A**). The AND gate **286** has one input coupled to the output of the inverter **278** (shown in FIG. **10B**). The remaining inputs of the AND gates **284** and **286** are coupled to the output of the XOR gate **270** (shown in FIG. **10B**) so that the mode signal is only encoded onto the red video signal when the vertical sync signal is activated.

The output of the AND gates **284** and **286** are coupled in series with a pair of resistors **290** and **292**, respectively. The resistors **290** and **292** are coupled together at a common node **291**. Connected between the node **291** and ground is a resistor **293**. Each time the vertical sync signal is active, the AND gates **284** and **286** produce a voltage at the node **291** that is proportional to the mode of the video monitor. The proportional voltage is fed into the inverting input of an amplifier **294**. The non-inverting input of the amplifier **294** is connected to receive the red analog video signal produced by the remote computer. When the vertical sync signal is in its normal state, the output signal of the comparator **294** follows the red analog video signal. However, when the vertical synchronize signal is activated, the mode signal is encoded on the red video signal.

After the video signals have been transmitted from the remote server computer and through the analog crosspoint switch to the remote workstation, the sync signals are extracted from the green and blue video signals. To extract the horizontal sync signal from the green video signal, the circuit shown in FIG. **1A** is used. The green video signal is received by the pod at a differential receiver **90** that produces an output signal which is fed to a non-inverting input of a clipping amplifier **302**. The output signal of the amplifier **302** is the green analog video signal that is fed to the video monitor. A resistor **306** is disposed between a non-inverting input of a comparator **304** to the output of the differential receiver **90**. Connected between a non-inverting output of the comparator **304** and the non-inverting input is a feedback resistor **308**. An inverting input of comparator **304** is tied to a constant reference voltage that is supplied by the voltage divider defined by resistors **310** and **312**. When the output signal of the differential receiver **90** has a magnitude below the voltage provided at the inverting input of the comparator **304**, the inverting output of amplifier **304** creates a positive going pulse. The positive going pulse is supplied to an input of an exclusive or (XOR) gate **314**. Coupled to another input of the exclusive or gate **314** is the horizontal mode (H-mode) signal which is recovered from the red analog video signal as will be described below. The XOR gate **314** adjusts the polarity of the horizontal sync signal depending on the value of the H-mode signal.

The circuit required to extract the vertical sync signal from the blue video signal is the same as the circuit shown in FIG. **11A** except that the exclusive or (XOR) gate receives the V-mode signal in order to adjust the polarity of the vertical sync signal.

To recover the video mode signal, the present invention utilizes the circuit shown in FIG. **11B**. The red analog video signal is received at a pod by a differential line receiver **90** that produces the red analog video signal. The output of the differential line receiver **90** is coupled to the inverting inputs of a pair of comparators **320** and **324**. The comparators **324** are gated by the output of a one shot **326** that is triggered by the rising edge of the vertical sync pulse so that the comparators only change state when the vertical sync signal is active. The noninverting input of comparator **324** is supplied with a reference voltage produced by a voltage divider that comprises a resistor **326** and a resistor **328**. The inverting input of the comparator **320** is supplied with a constant voltage produced by a voltage divider that comprises a resistor **330** and a resistor **332**.

A resistor **334** is placed between the output of comparator **320** and the inverting input of comparator **324**. Finally, a resistor **336** is placed between the inverting input of comparator **320** and the inverting input of comparator **324**.

The mode extract circuit produces two signals, H-mode and V-mode, having logic levels that are dependent on the magnitude of the mode signal encoded on the red video signal. If the magnitude of the mode signal is between 0 and −0.15 volts, the H-mode signal will be low and the V-mode signal will be low. When the mode signal has a magnitude between −0.15 and −0.29 volts, the H-mode signal will be high and the V-mode signal will remain low. The V-mode signal is high and the H-mode signal is low when the magnitude of the mode signal is between −0.29 volts and −0.49 volts. Both the H-mode and V-mode signals are high when the magnitude of the mode signal is less than −0.49 volts. As will be appreciated, the values given above will differ if different circuit components are used.

Once the video mode signal has been decoded from the red video signal, the values of H-mode and V-mode are used to adjust the polarity of the horizontal and vertical sync signals using the XOR gate shown in FIG. **11A**.

As can be seen, the circuits shown in FIGS. **10A–10C** and **11A, 11B** reduce the number of wires that must extend between the remote server computer and the workstation by encoding the sync and mode signals onto the color video signals at a time when the signals are normally unused.

Having now described the components of the present invention, its operation is described. To connect a workstation to a remote computer, a user sends a command that causes the central crosspoint switch to couple the keyboard/mouse signals to one of the remote computers. As indicated above, commands that affect the operation of the crosspoint switch as inserted between "printscreen" and "enter" keystrokes. The pod connected to the workstation detects these keys and transmits a packet to the CPU on one of the output cards. The CPU then transmits the packet to the master CPU that validates the request and issues a command to the switch cards to set the position of the 16×16 digital and analog switches **182** and **184** (FIG. **6**). Once the position of the switches has been set, the master CPU tells the computer pod **76** that the connection has occurred. The keyboard/mouse signals are then packetized and transmitted as pod to pod packets through the crosspoint switch. Video and audio signals from the remote computer are transmitted from the remote computer to the workstation.

As indicated above, the present invention provides the capability of allowing a user to send commands from a workstation to the central crosspoint switch in response to prompts that are displayed on the video monitor. The onscreen programming circuit **99** shown in FIG. **2** produces video signals that displays a menu of commands to be selected by the user. FIG. **12A** is a circuit diagram of the onscreen programming circuit **99**. The circuit includes a set of tri-state buffers **352**, **354** and **356** that have their inputs connected to the red, green and blue video signals provided by the sync extract circuit **94** (shown in FIG. **2**). When the tri-state buffers are energized, the red, green and blue video signals are passed to the video monitor. When the tri-state buffers **352**, **354** and **356** are in their high impedance state, the video signals are produced by an onscreen programming circuit **364**, as will be described.

The onscreen programming circuit **99** produces its own horizontal and vertical sync signals using a sync generator **358**. The horizontal and vertical sync signals produced are supplied to a switch **360** that selects either the sync signals produced by the internal sync generator **358** or the external horizontal and vertical sync signals recovered from the green and blue video signals transmitted from the remote computer. The switch **360** receives a signal on a lead **361** that is coupled to the CPU **80** (FIG. **2**) that determines which set of horizontal and vertical sync signals are selected. The horizontal and vertical sync signals selected by the switch **360** are fed to the video monitor at the user's workstation. Also connected to the output of the switch **360** is a sync polarizer **362** that forces the polarity of the horizontal and vertical sync signals selected to be active low. The details of the sync polarizer **362** are shown in FIG. **12B**.

The sync polarizer includes a pair of exclusive OR (XOR) gates **400** and **402**. The XOR gate **400** has one input connected directly to the sync signal to be polarized. A resistor **404** is connected between the sync signal and the other input of the XOR gate **400**. Connected between the second input of the XOR gate **400** and ground is a capacitor **406**. The voltage on the capacitor **406** is the average voltage of the sync signals. The output of the XOR gate **400** feeds an input of the XOR gate **402**. The other input of the XOR gate **402** is coupled to a logic high signal. The output of the XOR gate **402** will be a negative going pulse each time the sync signal is activated no matter what the normal state of the sync signal is.

The outputs of the sync polarizer **362** are coupled to a horizontal and vertical sync input of an onscreen processor **364**. The onscreen processor produces red, green and blue video signals that display one or more alphanumeric characters that are programmed in its internal video ROM memory. To dictate which characters are placed on the video screen, the CPU **80** generates serial I^{2}C interface signals on a pair of leads **363** and **365**. These signals are applied to the onscreen processor **364** which causes the processor to retrieve from an internal video RAM characters that are to be displayed on the video screen. The onscreen processor **364** provides two signals HBFK and HTONE that are supplied to an overlay control logic circuit **366**. Also supplied to the overlay control logic circuit are four signals from the CPU **80** of the user pod. These four signals are H Tone Enable, OSD Enable, System Video Enable and Transparent. The overlay control logic circuit **366** reads the value of these logic signals and either enables or disables a set of tri-state buffers **368**, **370** and **372** on the tri-state buffers **352**, **354** and **356**. These tri-state buffers **368**, **370** and **372** couple the outputs of the onscreen processor **364** to the leads that connect to the monitor's color inputs.

When the tri-state buffers **352**, **354** and **356** are in their high impedance state, and the tri-state buffers **368**, **370** and **372** are active, then the video screen will only display those signals produced by the onscreen processor. Conversely, if the tri-state buffers **368**, **370** and **372** are in their high impedance state and the tri-state buffers **352**, **354** and **356** are active then the monitor displays the video signals produced by the remote computer system. If both sets of tri-state buffers **368**, **370**, **372** and **352**, **354** and **356** are both active, then the monitor will display the video signals produced by both the onscreen processor and the remote computer system. The following is a table that defines the logic of the overlay control logic circuit **366**.

		H TONE	OSD		TRANS-	
HTONE	HBFK	ENABLE	ENABLE	SYS_VID_EN	PARENT	DISPLAY
X	0	X	0	0	X	screen blank
X	X	X	0	1	X	system video displayed only
X	1	0	1	0	0	OSD displayed only
1	1	1	1	0	0	OSD with transparent characters, i.e., characters, transparent, OSD windows solid
X	X	X	1	0	1	illegal state
0	1	0	1	1	0	active system video with solid OSD characters
1	1	1	1	1	0	active system video transparent OSD characters and solid OSD windows
1	1	0	1	1	1	active system video with opaque OSD characters and windows
1	1	1	1	1	1	active system video transparent OSD characters and opaque OSD windows

The construction of the overlay control logic circuit **366** given the above table is considered to be within the skill of an ordinary digital electronics engineer.

To activate the onscreen programming display, the user begins the escape sequence by pressing the "printscreen" key. The CPU within the user pod recognizes this key and produces a menu on the video screen. The user then selects one or more items from the menu by typing on the keyboard or moving the mouse. The CPU then interprets these mouse/keyboard inputs as commands that are to be transmitted to the central crosspoint switch. Once the user ends a command by activating the "enter" key, the CPU can generate one or more packets that are transmitted to the central crosspoint switch that enable the user to connect to a different computer, monitor the status of a different computer, etc.

As can be seen, the present invention allows a user to access any of thirty-two remotely located computers from a central workstation. The system operates apart from a network so that if the network fails, a user can still access each of the server computers. Furthermore, the pods act as translators between different keyboard/monitor types and different computers. Because all pod to pod packets have the same format, previously incompatible equipment can be easily coupled together.

While the preferred embodiment of the invention has been illustrated and described, it will be appreciated that various changes can be made therein without departing from the spirit and scope of the invention. For example, although the present invention is described with respect to connecting workstations to remotely located computers for the purposes of system administration, it will be appreciated that the invention also has further uses. For example, it may be desirable to locate expensive computer equipment away from relatively inexpensive terminals. Therefore, the present invention could be used in academic sessions where it is desirable to allow students to operate remotely located computers from one or more workstations. It is believed that the present invention has numerous applications where it is desirable to separate computing equipment from computer display and data input devices. Therefore, the scope of the invention is to be determined solely from the following claims.

The embodiments of the invention in which an exclusive property or privilege is claimed are defined as follows:

1. A switching system comprising:

a computer-side interface for simultaneously physically connecting to independent, dedicated cables of respective keyboard and analog video outputs of plural computers;

a user-side interface for physically connecting to a first set of independent, dedicated cables of a first keyboard and an analog video input of a first monitor;

an analog video receiving circuit, connected to the computer-side interface, for receiving analog video signals from one of the plural computers through the computer-side interface;

an analog video overlay image generating circuit, disposed between the computer-side interface and the user-side interface, for producing an analog overlay video signals internal to the switching system; and

an analog video overlay circuit, disposed between the computer-side interface and the user-side interface, for combining (1) a portion of the analog video signals received by the analog video receiving circuit and (2) the analog overlay video signals generated internally to the switching system to form a combined analog signal that is output to the first monitor via the user-side interface.

2. The system as claimed in claim **1**, wherein the analog video receiving circuit further comprises a synchronization signal detecting circuit for detecting one of a horizontal- and a vertical-synchronization signal corresponding to the analog video signals from the one of the plural computers.

3. The system as claimed in claim **1**, wherein the computer-side interface further comprises a computer-side mouse interface for transceiving signals to and from mouse ports of the plural computers, and

wherein the user-side interface further comprises a user-side mouse interface for transceiving signals to and from a first computer mouse.

4. The system as claimed in claim **1**, wherein the analog video overlay circuit comprises a multiplexer for multiplexing (1) the portion of the analog video signals received by the analog video receiving circuit and (2) the analog overlay video signals generated internally to the switching system without requiring a frame buffer.

5. The system as claimed in claim **1**, further comprising a keyboard translator disposed between the computer-side and the user-side interfaces for translating a keyboard code from a first format used on the user-side to a second format used on the computer-side.

6. The system as claimed in claim **1**, wherein the analog video receiving circuit receives separate analog red, green and blue signals.

7. The system as claimed in claim **2**, wherein the analog video receiving circuit comprises a receiving circuit for receiving real-time, analog video signals.

8. The system as claimed in claim **2**, wherein the analog video receiving circuit comprises a receiving circuit for receiving real-time, analog video signals including the at least one of the horizontal- and a vertical-synchronization signal superimposed on the real-time, analog video signals.

9. The system as claimed in claim **7**, wherein the analog video receiving circuit receives separate analog red, green and blue signals.

10. The system as claimed in claim **8**, wherein the analog video receiving circuit receives separate analog red, green and blue signals.

11. The system as claimed in claim 1, further comprising:

a digital backplane; and

an analog backplane, wherein keyboard information is routed from the computer-side interface to the user-side interface on the digital backplane independent of the analog video signals that are routed from the computer-side interface to the user-side interface on the analog backplane.

12. The system as claimed in claim **1**, wherein the computer-side interface comprises a parallel switch for connecting the plural computers to the computer-side interface in parallel.

13. The system as claimed in claim **1**, wherein the user-side interface further comprises a secondary user-interface for connecting to a second set of independent, dedicated cables of a second keyboard and an analog video input of a second monitor, wherein the analog video signals of any one of the plural computers can be routed simultaneously to both the first and second monitors in parallel.

14. The system as claimed in claim **1**, further comprising a keyboard command detector, disposed between the computer-side interface and the user-side interface, for detecting when the first keyboard requests that the analog video signals of the one of the plural computers be replaced by analog video signals of another of the plural computers.

15. The system as claimed in claim **14**, further comprising:

a digital backplane; and

an analog backplane, wherein keyboard information is routed from the computer-side interface to the user-side interface on the digital backplane independent of the analog video signals that are routed from the computer-side interface to the user-side interface on the analog backplane.

16. The system as claimed in claim **1**, further comprising a keyboard command detector, disposed between the computer-side interface and the user-side interface, for detecting when the first keyboard requests that the analog video signals of the one of the plural computers be replaced by analog video signals of another of the plural computers in response to the combined analog signal that is output to the first monitor via the user-side interface.

17. The system as claimed in claim **16**, further comprising:

a digital backplane; and

an analog backplane, wherein keyboard information is routed from the computer-side interface to the user-side interface on the digital backplane independent of the analog video signals that are routed from the computer-side interface to the user-side interface on the analog backplane.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 6,112,264 Page 1 of 1
DATED : August 29, 2000
INVENTOR(S) : Danny L. Beasley et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 8,
Line 12, change "CD22M34945Q" to -- CD22M3494SQ --.

Signed and Sealed this

Nineteenth Day of November, 2002

Attest:

JAMES E. ROGAN
Director of the United States Patent and Trademark Office

Attesting Officer

US006345323B1

(12) **United States Patent**
Beasley et al.

(10) **Patent No.: US 6,345,323 B1**
(45) **Date of Patent: *Feb. 5, 2002**

(54) **COMPUTER INTERCONNECTION SYSTEM**

(75) Inventors: **Danny L. Beasley**, Mukilteo; **Robert V. Seifert, Jr.**, Redmond; **Paul Lacrampe**, Seattle; **James C. Huffington**, Edmond; **Thomas Greene**, Bellevue; **Kevin J. Hafer**, Woodinville, all of WA (US)

(73) Assignee: **Apex, Inc.**, Redmond, WA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **09/590,170**

(22) Filed: **Jun. 9, 2000**

Related U.S. Application Data

(63) Continuation of application No. 09/244,947, filed on Feb. 4, 1999, now Pat. No. 6,112,264, which is a continuation of application No. 08/969,723, filed on Nov. 12, 1997, now Pat. No. 5,884,096, which is a continuation of application No. 08/519,193, filed on Aug. 25, 1995, now Pat. No. 5,721,842.

(51) **Int. Cl.**^{7} **G06F 13/14**; G06F 3/153
(52) **U.S. Cl.** **710/38**; 709/208
(58) **Field of Search** 709/209, 208; 710/15, 38, 2, 51, 131

(56) **References Cited**

FOREIGN PATENT DOCUMENTS

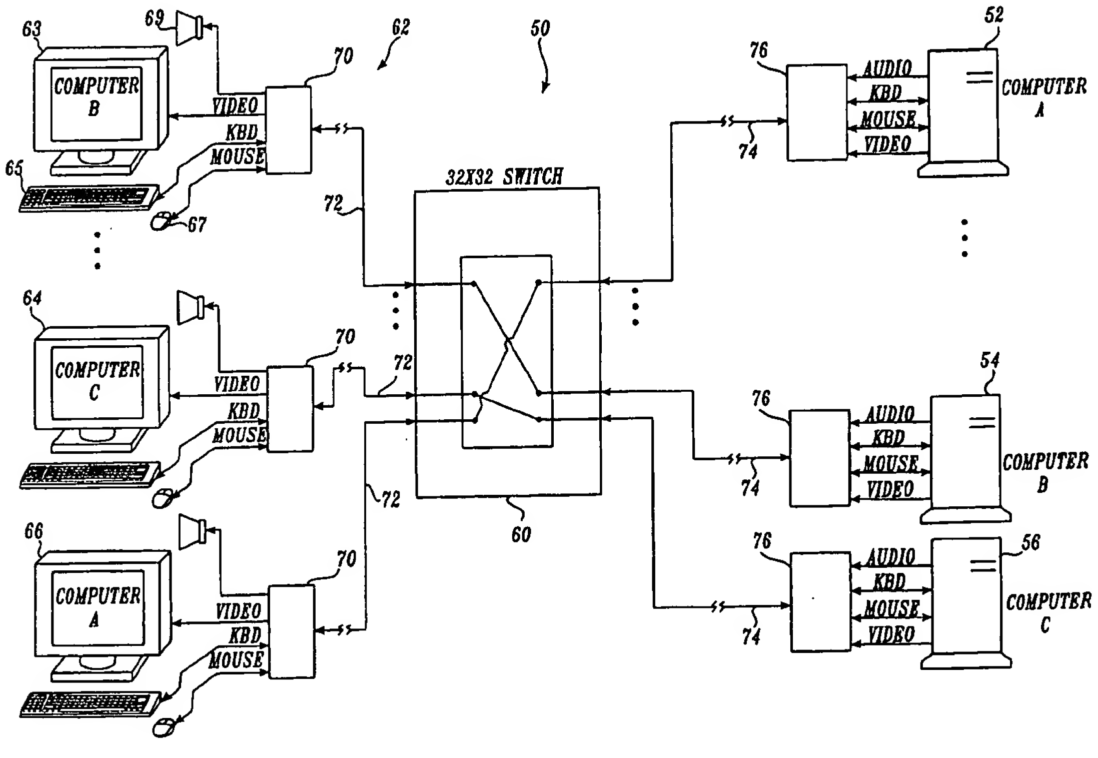
WO	94/19749	9/1944

Primary Examiner—Dung C. Dinh
(74) *Attorney, Agent, or Firm*—Oblon, Spivak, McClelland, Maier & Neustadt, P.C.

(57) **ABSTRACT**

A computerized switching system for coupling a workstation to a remotely located computer. A signal conditioning unit receives keyboard and mouse signals generated by a workstation and generates a data packet which is transmitted to a central crosspoint switch. The packet is routed through a crosspoint switch to another signal conditioning unit located at a remotely located computer. The second signal conditioning unit applies the keyboard and mouse commands to the keyboard and mouse connectors of the computer as if the keyboard and mouse were directly coupled to the remote computer. Video signals produced by the remote computer are transmitted through the crosspoint switch to the workstation. Horizontal and vertical sync signals are encoded on to the video signals to reduce the number of cables that extend between the workstation and the remote computer. The signal conditioning units connected to the workstations include an onscreen programming circuit that produces menus for the user on a video display of the workstation.

8 Claims, 13 Drawing Sheets



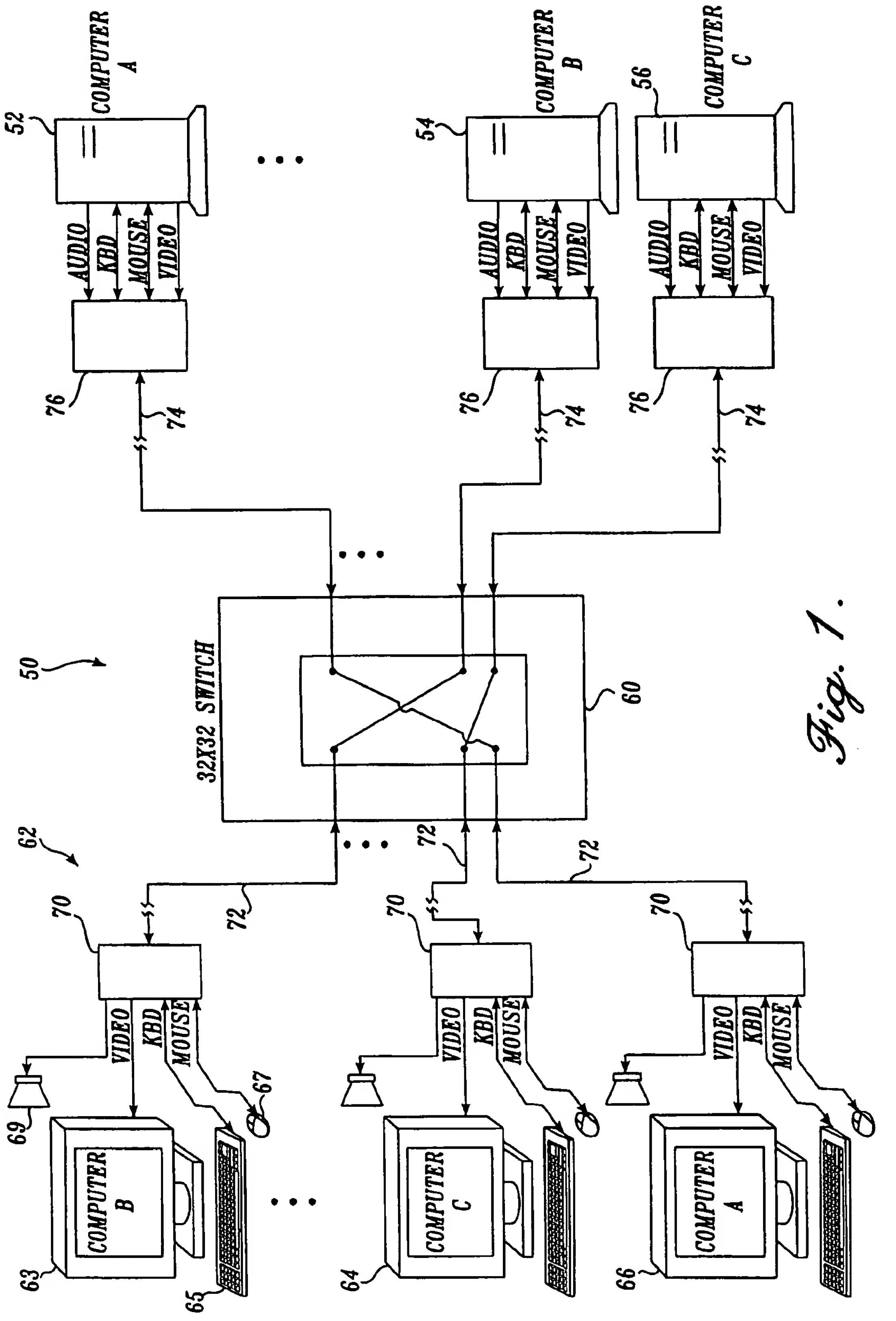

52
COMPUTER A
54
COMPUTER B
56
COMPUTER C
AUDIO
KBD
MOUSE
VIDEO
76
74
50
32X32 SWITCH
60
62
72
70
VIDEO
KBD
MOUSE
69
67
63
65
64
66
COMPUTER B
COMPUTER C
COMPUTER A

Fig. 1.

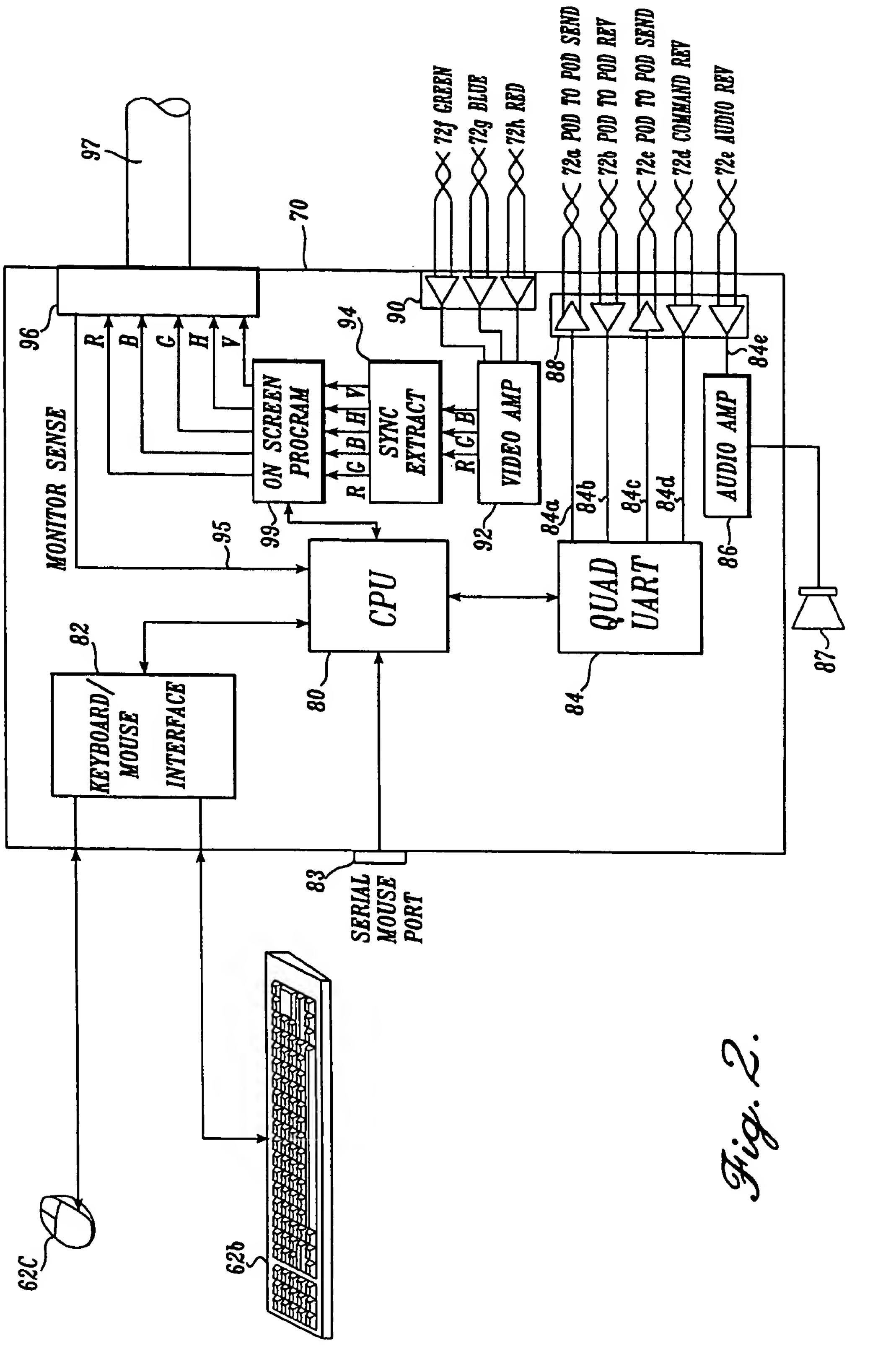

97
70
72f GREEN
72g BLUE
72h RED
72a POD TO POD SEND
72b POD TO POD REV
72c POD TO POD SEND
72d COMMAND REV
72e AUDIO REV
96
R
B
G
H
V
94
90
88
84e
MONITOR SENSE
ON SCREEN PROGRAM
SYNC EXTRACT
VIDEO AMP
AUDIO AMP
R G B H V
R G B
84a
84b
84c
84d
95
99
92
86
CPU
QUAD UART
87
82
KEYBOARD/ MOUSE INTERFACE
80
84
83
SERIAL MOUSE PORT
62C
62b

Fig. 2.

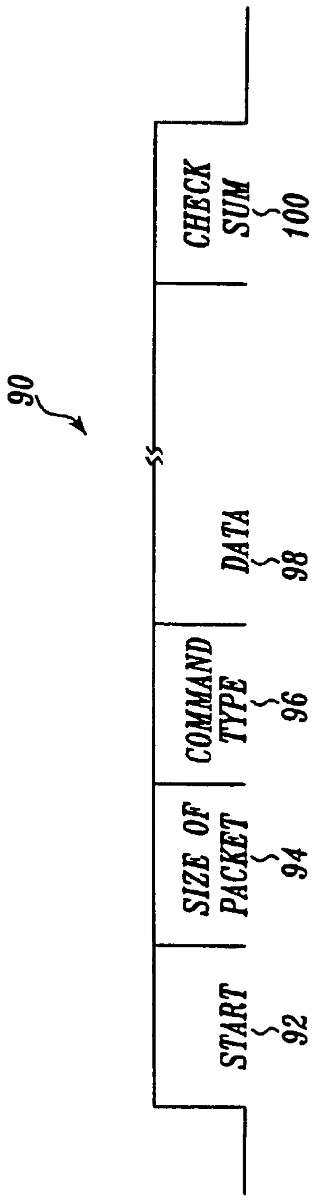

90
START
92
SIZE OF PACKET
94
COMMAND TYPE
96
DATA
98
CHECK SUM
100

Fig. 2A.

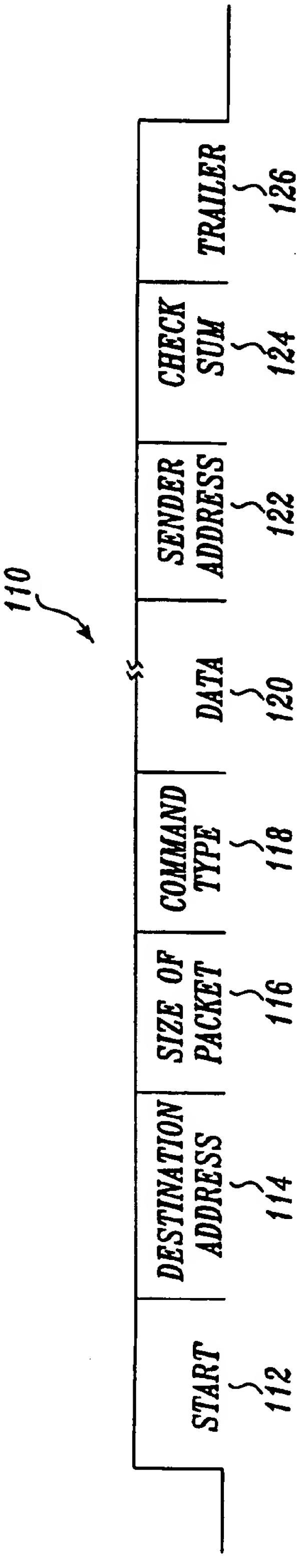
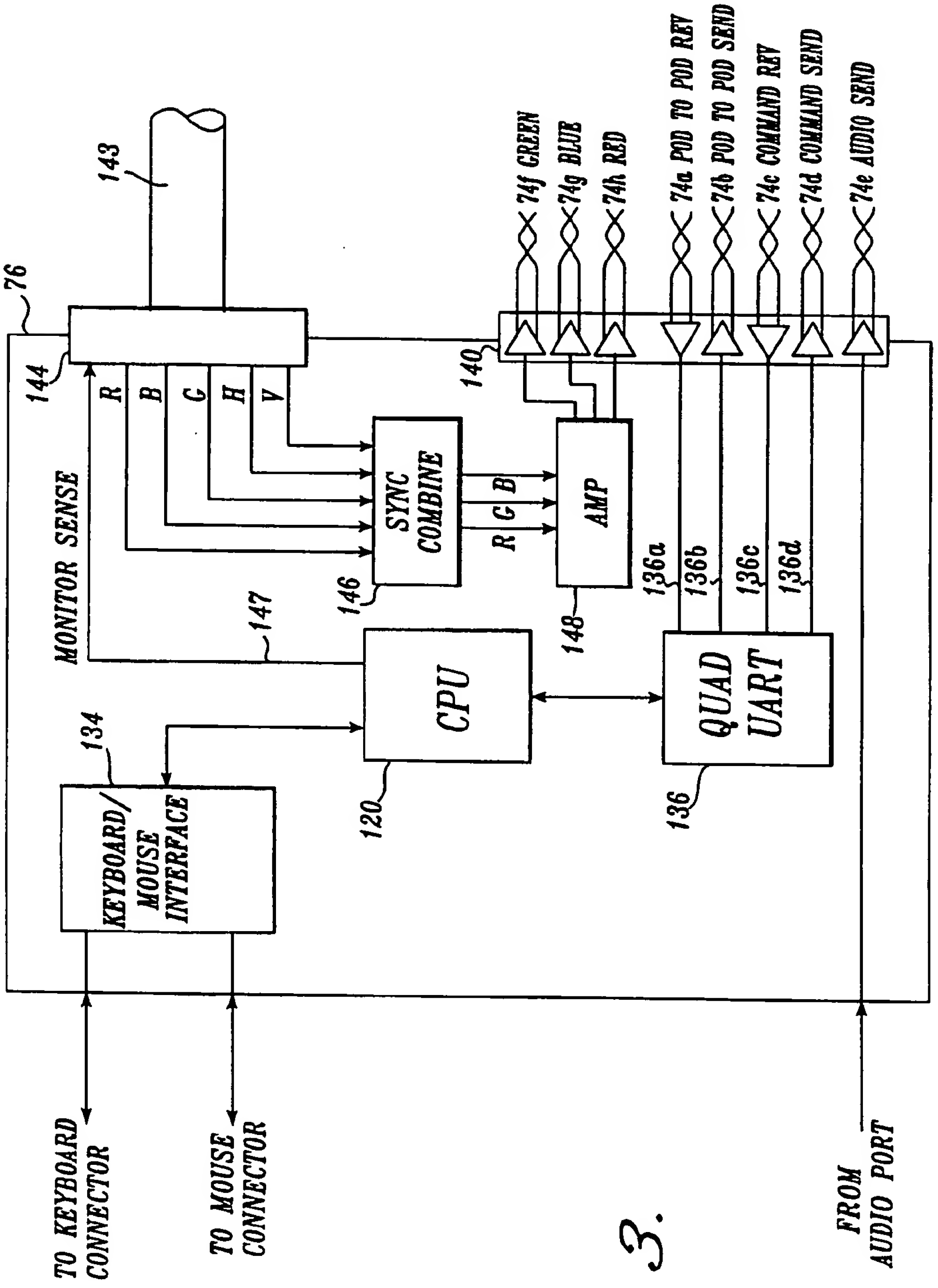

110
START
112
DESTINATION ADDRESS
114
SIZE OF PACKET
116
COMMAND TYPE
118
DATA
120
SENDER ADDRESS
122
CHECK SUM
124
TRAILER
126

Fig. 2B.



143
76
144
R
B
G
H
V
MONITOR SENSE
147
SYNC COMBINE
146
R G B
AMP
148
140
74f GREEN
74g BLUE
74h RED
74a POD TO POD REV
74b POD TO POD SEND
74c COMMAND REV
74d COMMAND SEND
74e AUDIO SEND
136a
136b
136c
136d
CPU
120
QUAD UART
136
KEYBOARD/ MOUSE INTERFACE
134
TO KEYBOARD CONNECTOR
TO MOUSE CONNECTOR
FROM AUDIO PORT

Fig. 3.

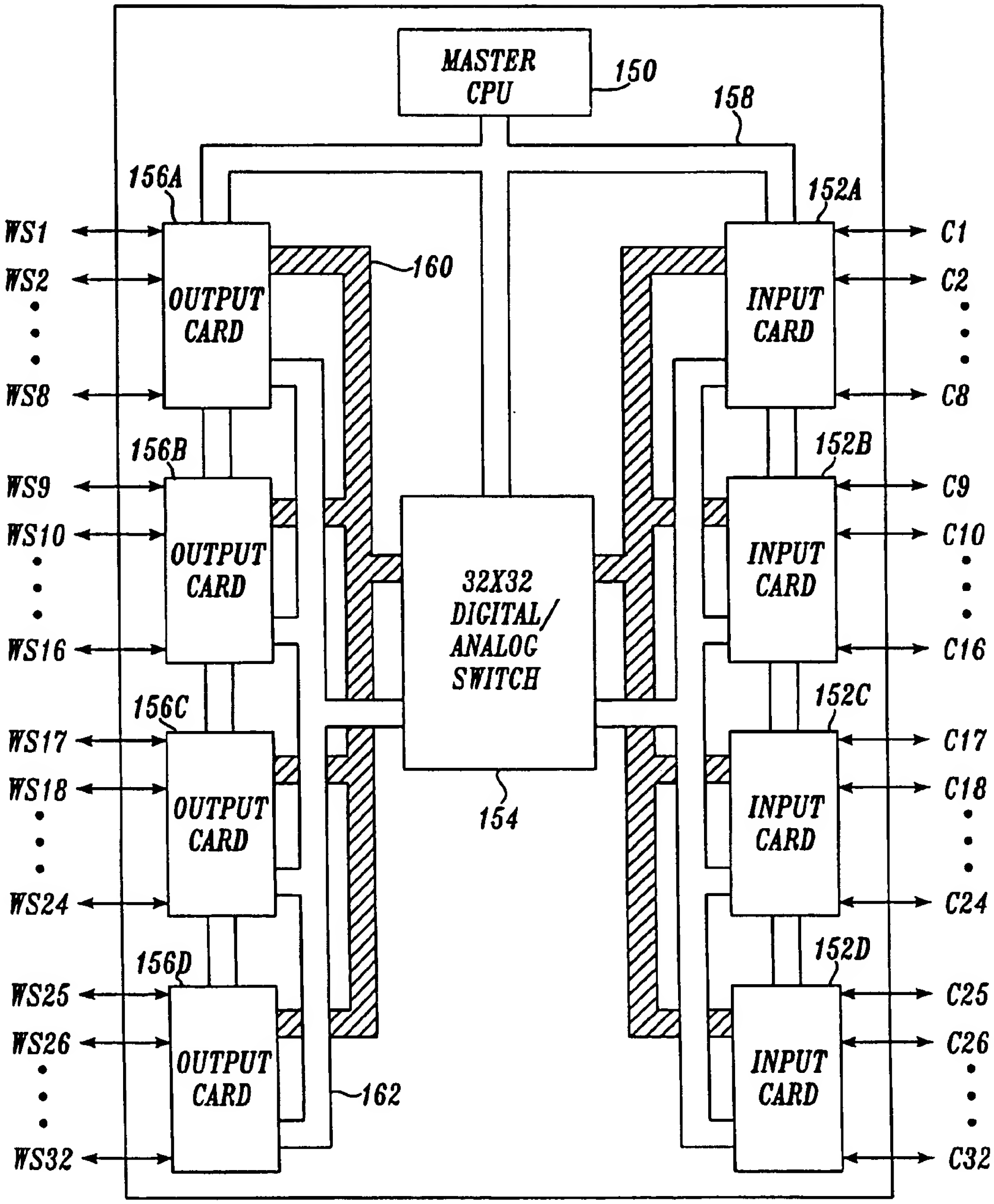

MASTER CPU
150
158
156A
152A
WS1
WS2
WS8
OUTPUT CARD
160
INPUT CARD
C1
C2
C8
156B
152B
WS9
WS10
WS16
OUTPUT CARD
32X32 DIGITAL/ ANALOG SWITCH
INPUT CARD
C9
C10
C16
156C
152C
WS17
WS18
WS24
OUTPUT CARD
154
INPUT CARD
C17
C18
C24
156D
152D
WS25
WS26
WS32
OUTPUT CARD
162
INPUT CARD
C25
C26
C32

Fig. 4.

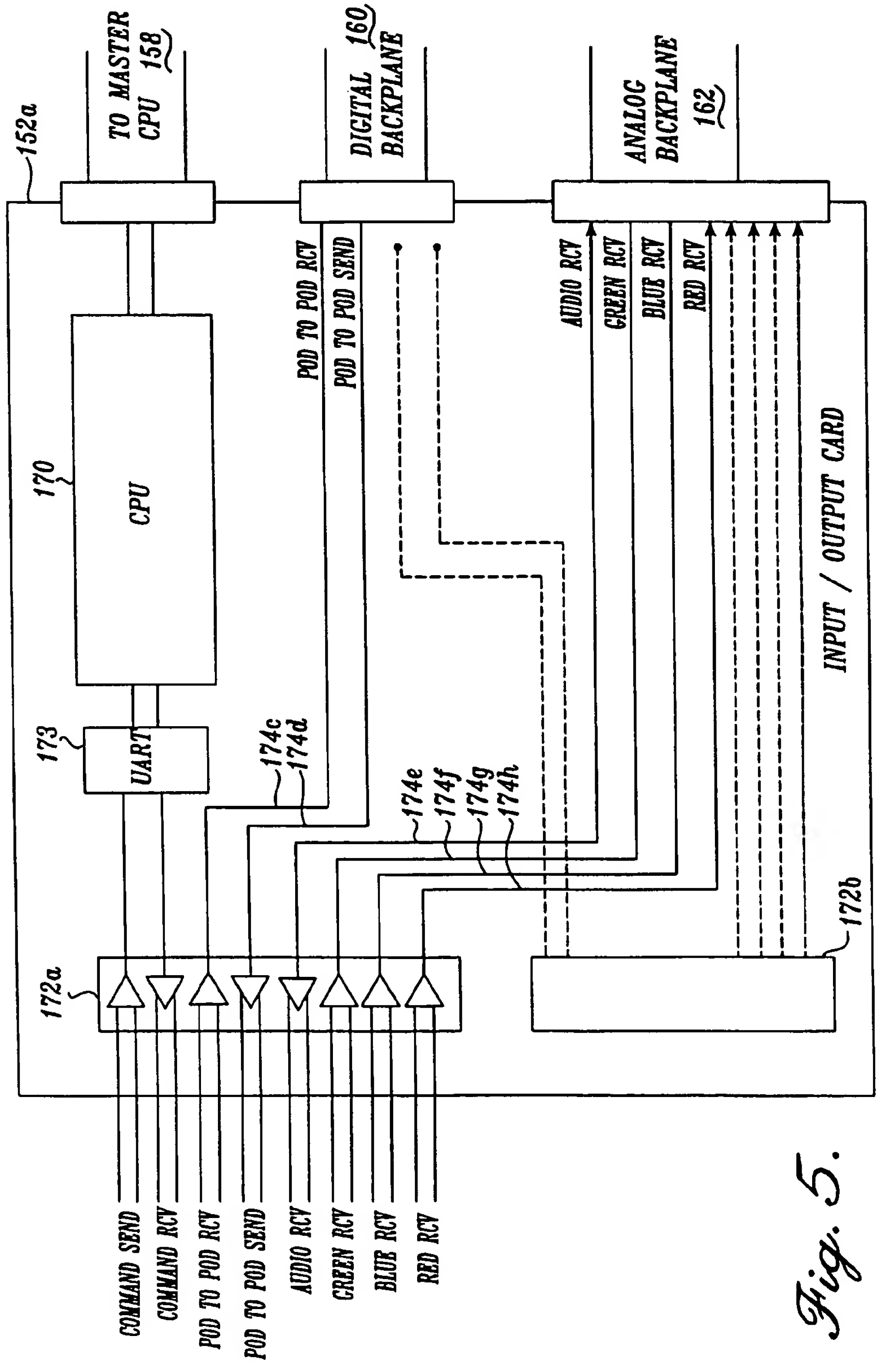

152a
TO MASTER CPU 158
DIGITAL 160 BACKPLANE
ANALOG BACKPLANE 162
POD TO POD RCV
POD TO POD SEND
AUDIO RCV
GREEN RCV
BLUE RCV
RED RCV
170
CPU
173
UART
174c
174d
174e
174f
174g
174h
INPUT / OUTPUT CARD
172a
172b
COMMAND SEND
COMMAND RCV
POD TO POD RCV
POD TO POD SEND
AUDIO RCV
GREEN RCV
BLUE RCV
RED RCV

Fig. 5.

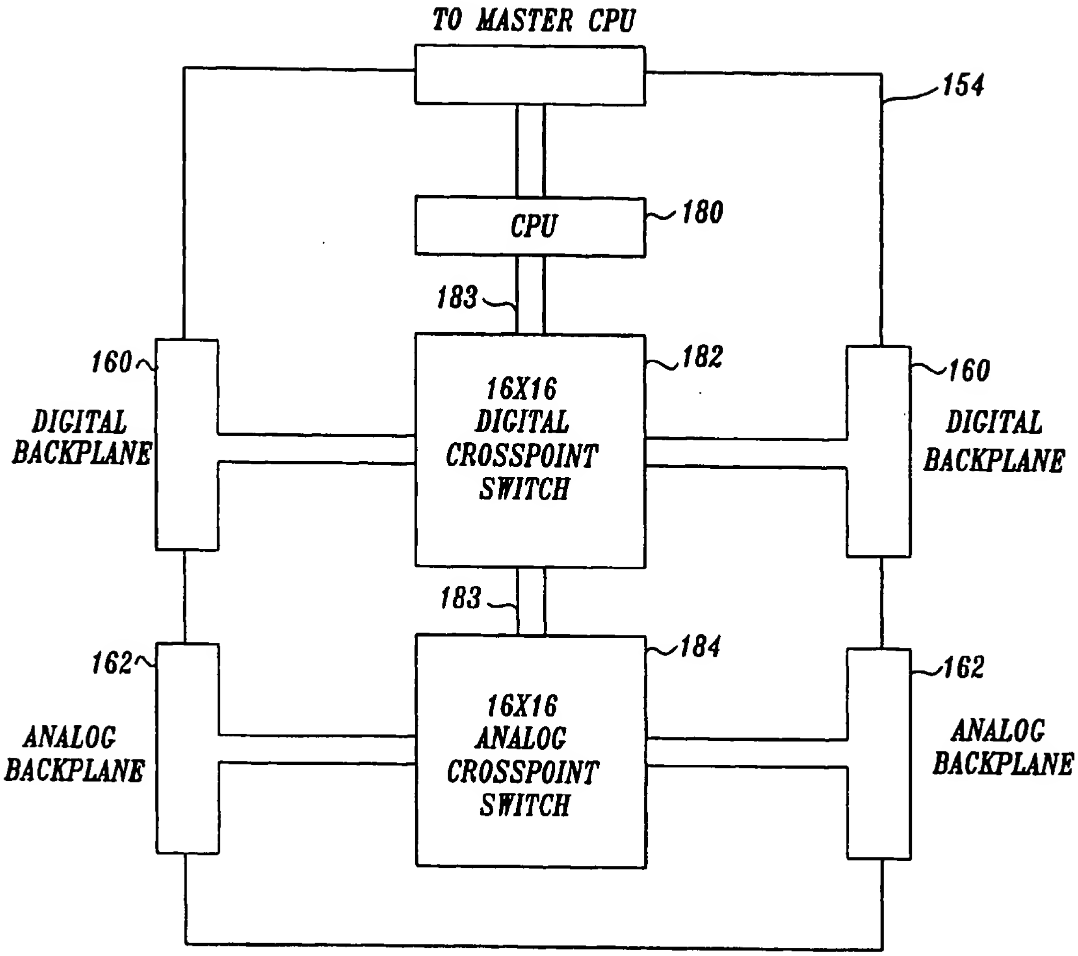

TO MASTER CPU
154
CPU
180
183
182
160
DIGITAL BACKPLANE
16X16 DIGITAL CROSSPOINT SWITCH
160
DIGITAL BACKPLANE
183
184
162
ANALOG BACKPLANE
16X16 ANALOG CROSSPOINT SWITCH
162
ANALOG BACKPLANE

Fig. 6.

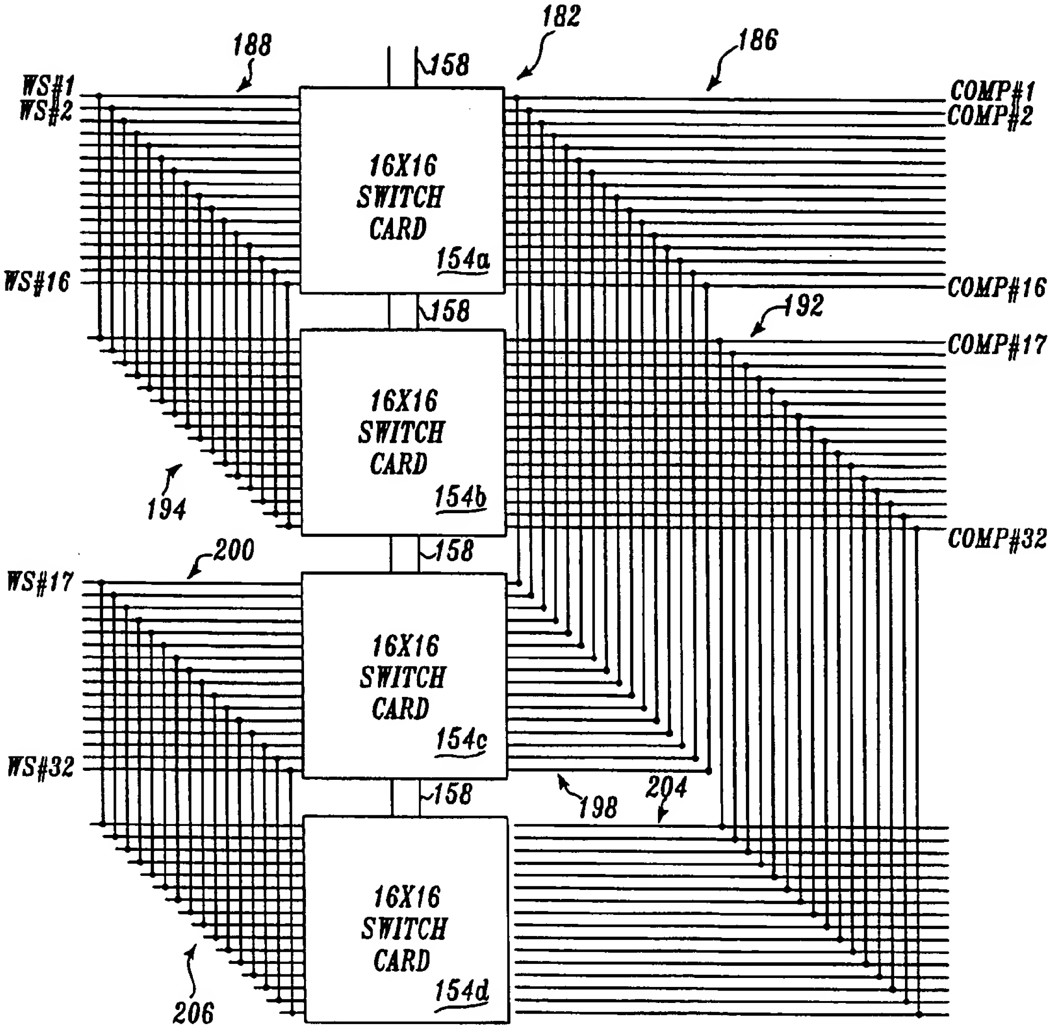

188
158
182
186
WS#1
WS#2
COMP#1
COMP#2
16X16
SWITCH
CARD
154a
WS#16
COMP#16
158
192
COMP#17
16X16
SWITCH
CARD
154b
194
COMP#32
158
200
WS#17
16X16
SWITCH
CARD
154c
WS#32
198
204
158
16X16
SWITCH
CARD
154d
206

Fig. 7.

Fig. 8.

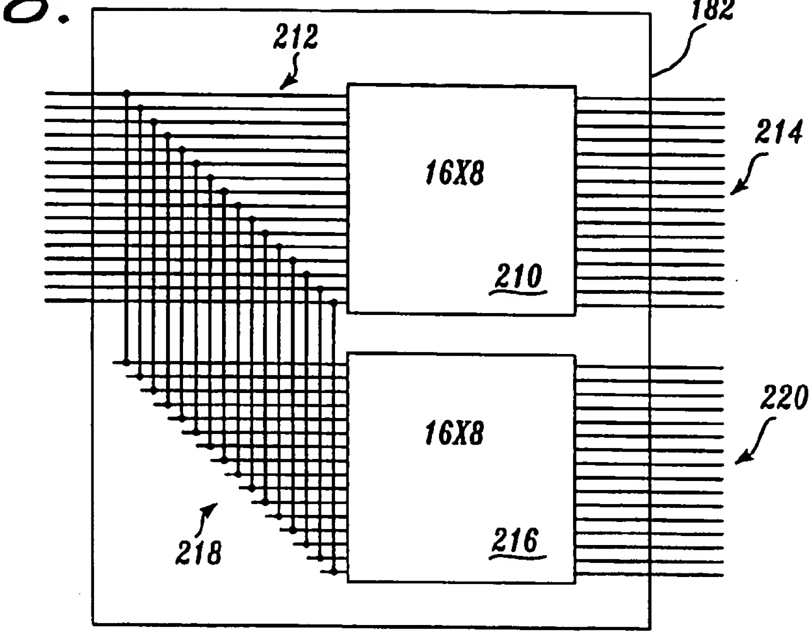
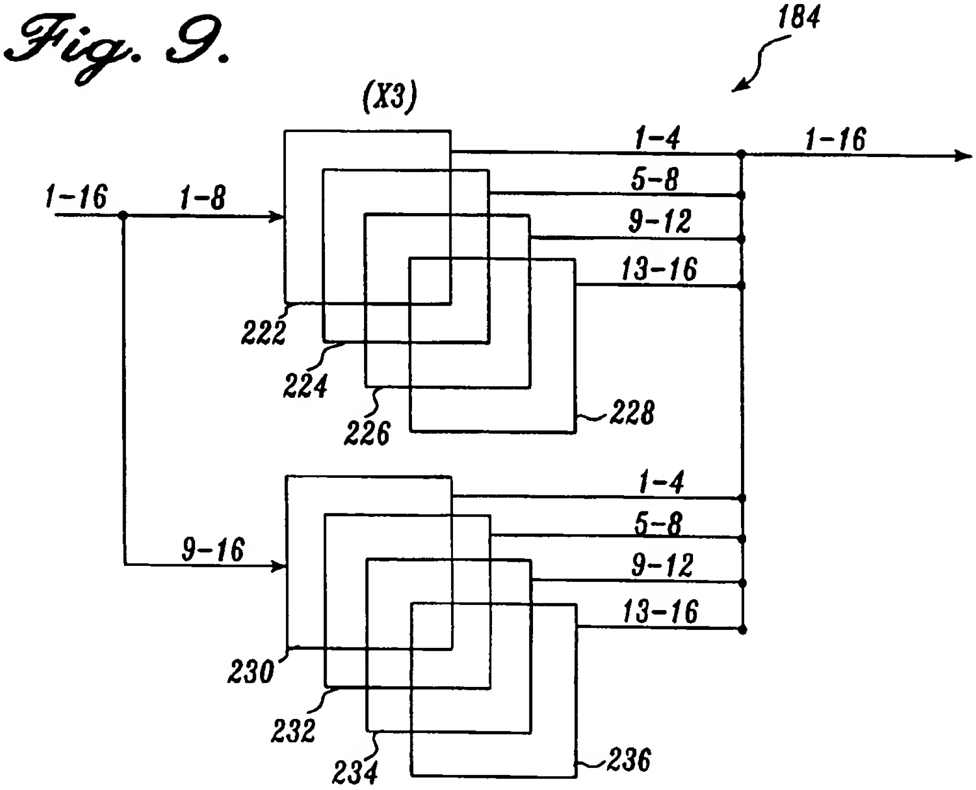


Fig. 9.



THE VIDEO BUS USES 8 8X4 SWITCHES PER COLOR.
TOTAL VIDEO SWITCHES EQUALS 24.

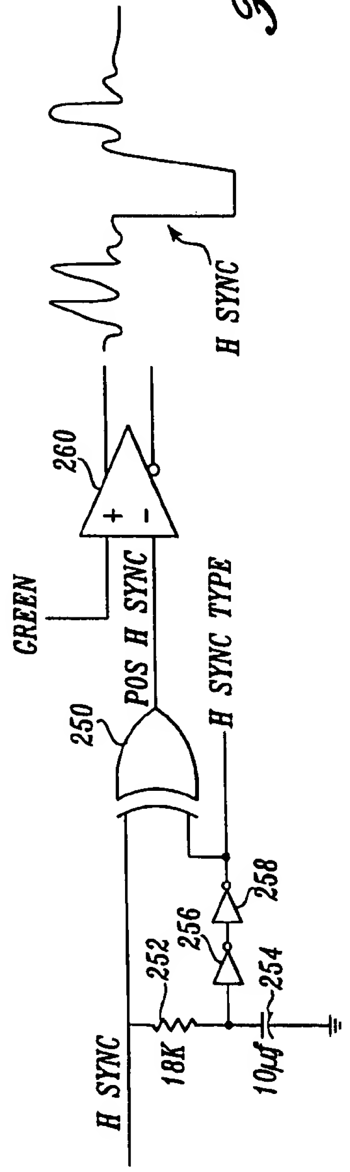


Fig. 10A.

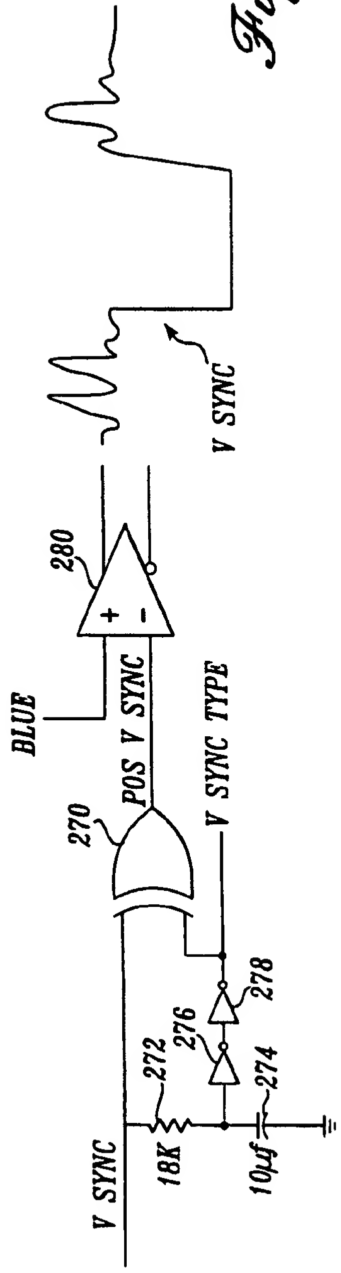


Fig. 10B.

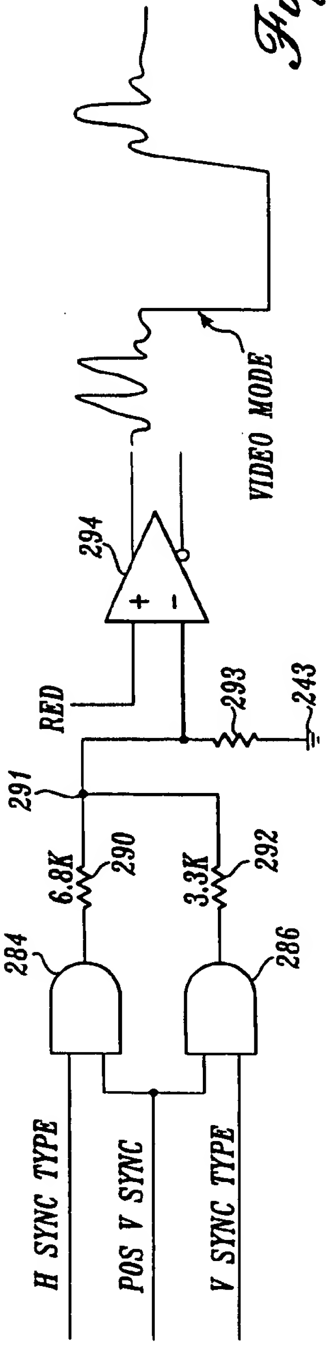


Fig. 10C.

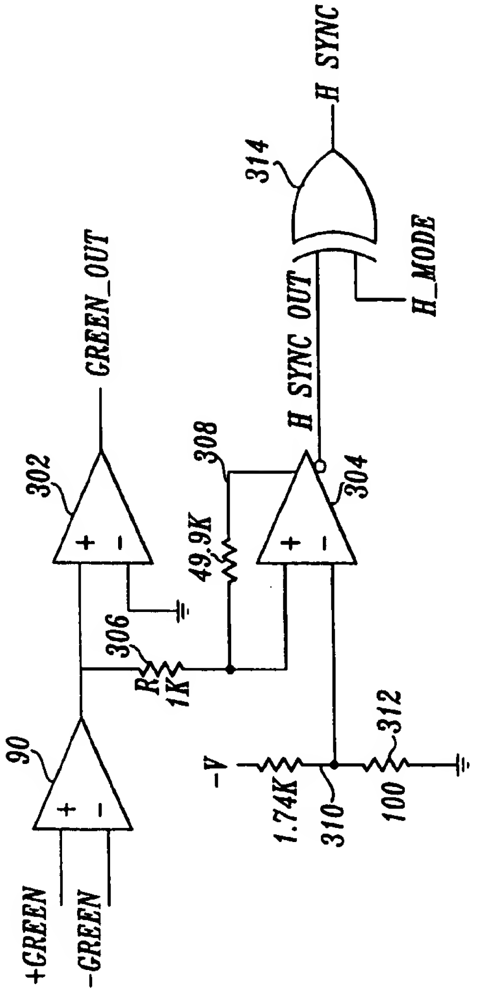


Fig. 11A.

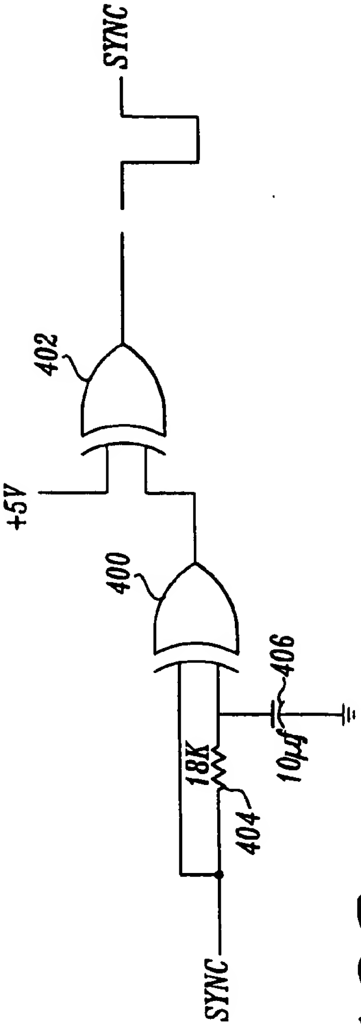


Fig. 12B.

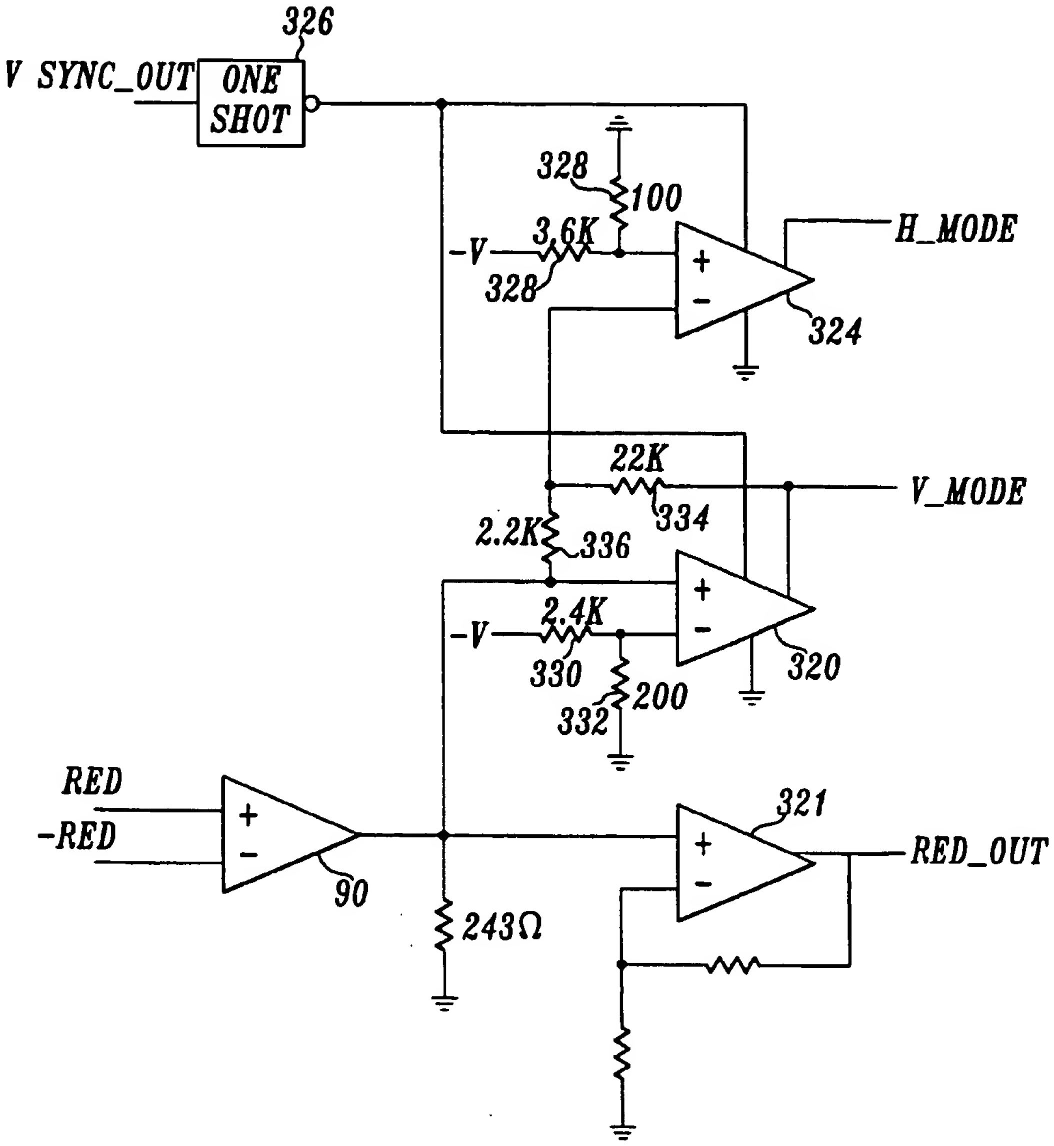

326
V SYNC_OUT
ONE SHOT
328
100
3.6K
-V
328
H_MODE
324
22K
V_MODE
334
2.2K
336
2.4K
-V
330
200
332
320
RED
-RED
90
243Ω
321
RED_OUT

Fig. 11B.

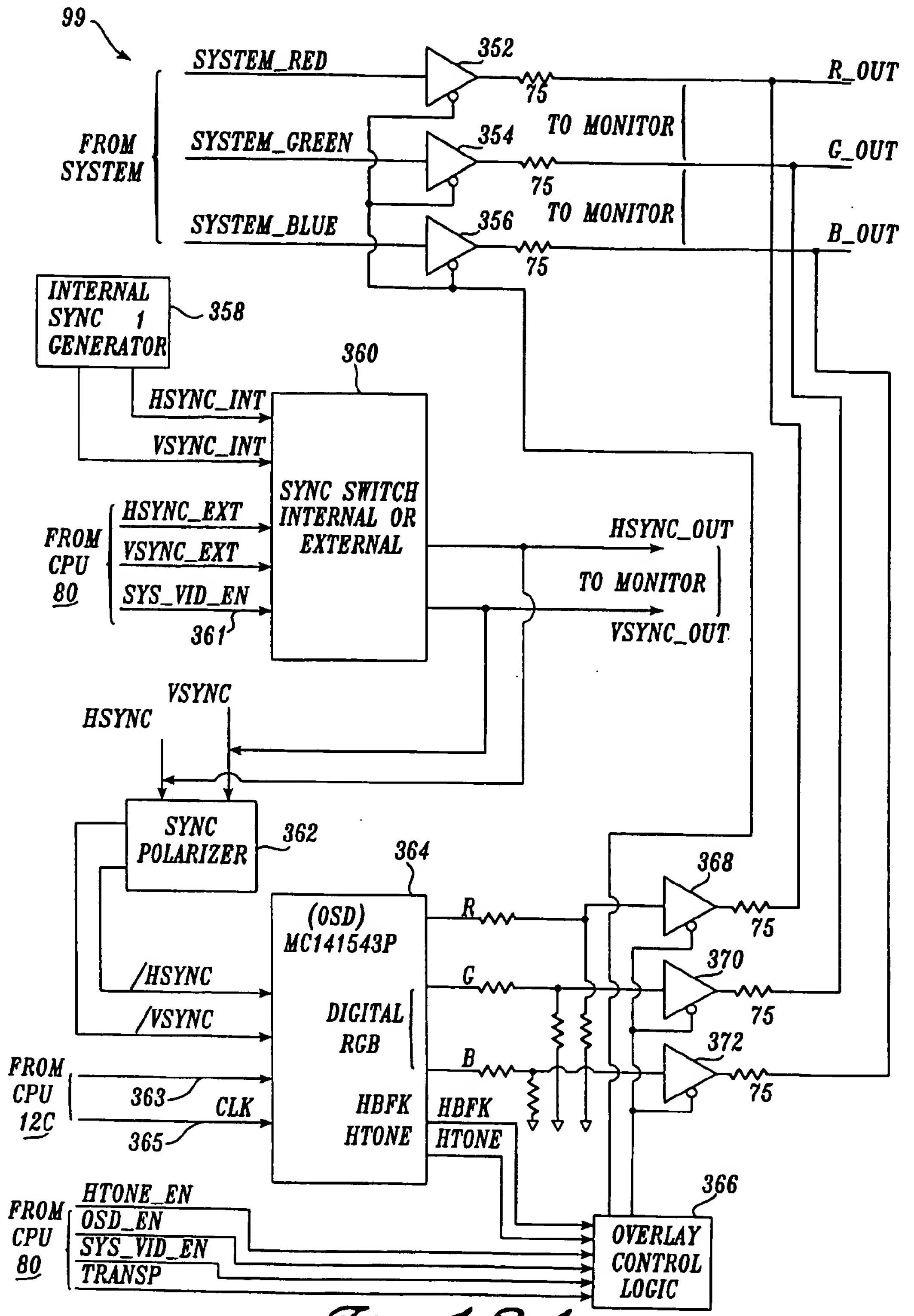

99
SYSTEM_RED
352
R_OUT
75
TO MONITOR
FROM SYSTEM
SYSTEM_GREEN
354
G_OUT
75
TO MONITOR
SYSTEM_BLUE
356
B_OUT
75
INTERNAL SYNC 1 GENERATOR
358
360
HSYNC_INT
VSYNC_INT
SYNC SWITCH INTERNAL OR EXTERNAL
FROM CPU 80
HSYNC_EXT
VSYNC_EXT
SYS_VID_EN
361
HSYNC_OUT
TO MONITOR
VSYNC_OUT
VSYNC
HSYNC
SYNC POLARIZER
362
364
368
(OSD) MC141543P
R
75
/HSYNC
G
370
75
/VSYNC
DIGITAL RGB
B
372
75
FROM CPU 12C
363
CLK
365
HBFK
HTONE
HBFK
HTONE
366
FROM CPU 80
HTONE_EN
OSD_EN
SYS_VID_EN
TRANSP
OVERLAY CONTROL LOGIC

Fig. 12A.

COMPUTER INTERCONNECTION SYSTEM

This application is a Continuation of application Ser. No. 09/244,947 filed on Feb. 4, 1999 now U.S. Pat. No. 6,112,264, which is a continuation of Ser. No. 08/969,723, filed Nov. 12, 1997, now U.S. Pat. No. 5,884,096, which is a continuation of Ser. No. 08/519,193, now U.S. Pat. No. 5,721,842.

FIELD OF THE INVENTION

The present invention relates to systems for interconnecting remotely located computers.

BACKGROUND OF THE INVENTION

In a typical local computer network there are a number of client computers that are coupled via a communication link to a number of network server resources. These resources include file servers, print servers, modem servers, and CD-ROM servers for example. Each server is usually a stand alone computer with its own keyboard, mouse and video monitor. Each client computer can utilize the functions provided by the server computers through the communication link.

Most computer networks have one or more system administrators, i.e. human operators, for the server computers. The system administrators monitor the operation of the software running on the server computers, load new software packages, delete outdated files and perform other tasks necessary to maintain the operation of the network. While most administrator tasks (modifying software, deleting files, etc.) can be performed over the network from a client computer, there are some situations where the network administrators must be physically located at the server computers for direct access to and operation of them. For example, it is not possible to reboot a server computer over the network. If the server computers are not close together, the time required for a task as simple as rebooting can be substantial.

Although it is possible to run dedicated communication links to each server computer in order to allow a system administrator to operate the network from a central location, a large number of cables are required for anything other than a very simple network.

SUMMARY OF THE INVENTION

The present invention provides a computerized switching system that allows centrally located network administrators to operate multiple server computers over long distances without requiring a complicated wiring scheme. In general, the switching system allows data transmission between a workstation and a remotely located server computer. A signal conditioning unit receives keyboard and mouse signals from a workstation and generates a serial data packet which is transmitted to a central crosspoint switch. The crosspoint switch routes the keyboard/mouse packet to another signal conditioning unit that is coupled to the remotely located server computer. The signal conditioning unit coupled to the server computer decodes the keyboard/mouse packet and applies the signals to a keyboard and mouse connector on the remote computer in the same manner as if the mouse and keyboard were directly coupled to the remote computer.

Video signals produced by the remote computer are transmitted through the crosspoint switch to the workstation. In order to minimize the number of wires extending between the remote computer and the workstation, the horizontal and vertical sync signals as well as a mode signal are encoded with the analog video signals. The present embodiment of the invention allows any of thirty-two workstations to be connected to any of thirty-two remotely located server computers.

BRIEF DESCRIPTION OF THE DRAWINGS

The foregoing aspects and many of the attendant advantages of this invention will become more readily appreciated as the same becomes better understood by reference to the following detailed description, when taken in conjunction with the accompanying drawings, wherein:

FIG. **1** is a pictorial diagram of a computerized switching system, according to the present invention, a number of workstations and a number of remotely-located computers;

FIG. **2** is a block diagram of a signal conditioning unit (pod) that is coupled to a workstation;

FIG. **2A** is a timing diagram of a serial pod to pod packet that is transmitted by the signal conditioning unit shown in FIG. **2**;

FIG. **2B** is a timing diagram of a data packet that is routed within the central crosspoint switch;

FIG. **3** is a block diagram of a signal conditioning unit (pod) that is coupled to a remote computer system;

FIG. **4** is a block diagram of a crosspoint switch according to the present invention that routes data between a workstation and a remote server computer;

FIG. **5** is a block diagram of an input/output card that is utilized to send and receive signals at the crosspoint switch;

FIG. **6** is a block diagram of a switch card that routes signals through the crosspoint switch;

FIG. **7** is a schematic diagram showing the interconnection of four switch cards to create a 32x32 switch utilized in the crosspoint switch of the present invention;

FIGS. **8** and **9** are schematic diagrams showing how a digital and analog 16x16 switch is constructed;

FIGS. **10A–10C** are schematic diagrams of circuits for encoding horizontal sync, vertical sync and video mode signals onto an analog video signal according to another aspect of the present invention;

FIGS. **11A** and **11B** are schematic diagrams of circuits for extracting the encoded horizontal and vertical sync signals and the mode signal from an analog video signal;

FIG. **12A** is a circuit diagram of an onscreen programming circuit that produces video displays on the workstation's monitor according to yet another aspect of the present invention; and

FIG. **12B** is a circuit diagram of a circuit that inverts the polarity of horizontal and vertical sync signals that is used within the onscreen programming circuit of FIG. **12A**.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

The present invention is a computerized switching system for allowing a number of computer workstations to be coupled to a number of remotely-located server computers. In the presently preferred embodiment of the invention, up to thirty-two workstations can be connected to any of thirty-two remote computer systems. However, those skilled in the art will recognize that the number of possible interconnections can easily be modified for the environment in which the invention is to be used.

Referring now to FIG. **1**, the computerized switching system or crosspoint switch according to the present invention allows a number of server computers 52, 54, 56 to be coupled to a number of workstations 62, 64, 66. Each workstation includes a video monitor 63, a keyboard 65 and a cursor control device such as a mouse 67. In accordance with the present invention, signals from the keyboard 65 and the mouse 67 are received by a signal conditioning circuit or pod 70. The pod transmits the keyboard and mouse signals over a communication link 72 to a central crosspoint switch 60. After being routed through the crosspoint switch 60, the keyboard and mouse signals are retransmitted on another communication link 74 to a pod 76, which is coupled to the remotely-located server computer. The pod 76 supplies the keyboard and mouse signals through appropriate connectors to keyboard and mouse input ports of the remote computer, just as if the keyboard 65 and mouse 67 were directly coupled to the keyboard and mouse input ports.

Audio and video signals produced by the remote server computer 52, 54 or 56 are received by the associated pod 76 and transmitted in the reverse direction along the communication link 74 to the central crosspoint switch 60. The central crosspoint switch routes the audio and video signals to one of the communication links 72 for transmission to a pod 70. The pod 70 then supplies the audio and video signals to the associated video monitor 63 and a speaker 69 of the workstation. From a user's perspective, the work station appears as if it is directly coupled to the remote server computer.

FIG. 2 is a block diagram of a pod 70. As described above, the pod operates to receive the mouse and keyboard signals and to transmit them through the crosspoint switch to a remotely-located server computer system. In addition, the pod receives video and audio signals from the remote server computer by way of the central crosspoint switch and supplies them to the video monitor and speaker of the workstation.

The pod 70 generally comprises a central processing unit (CPU) 80 having its own random access and read only memories. A keyboard/mouse interface 82 is coupled to the CPU 80 to receive and condition the electronic signals from the keyboard 65 and mouse 67. As the user moves the mouse or types on the keyboard, the keyboard/mouse interface 82 generates an interrupt signal that is fed to the CPU 80. The CPU 80 then reads the digitally buffered keyboard and mouse signals from the keyboard/mouse interface 82 and converts the signals into a data packet that is transmitted to the remote computer.

As shown in FIG. 2A, the pod to pod data packet 90 begins with a unique character 92 that marks the beginning of the data packet followed by a byte 94 that indicates the length of the packet. The next byte 96 identifies the type of data (mouse, keyboard, monitor type etc.) that the packet represents. The next series of bytes 98 represents the keyboard/mouse data to be transmitted to the server computer. Finally, a checksum byte 100 allows for the correction of errors that may occur during transmission.

It should be noted that the pod to pod packets are not limited to carrying keyboard and mouse data. The packets allow the pod at the work station to "talk to" the pod at the remote computers. Each pod acknowledges to the other that a packet was received correctly and in case of an error requests that a packet be retransmitted.

After the CPU 80 has assembled the pod to pod packet, the packet is transmitted to a quad UART 84, which transmits and receives serial data on four leads 84*a*–84*d*. The pod to pod packet is serialized and transmitted on the lead 84*a* to a differential line driver/receiver 88 that transmits and receives data on a number of twisted-pair cables 72*a*–72*e*, that are coupled to the central crosspoint switch 60 (shown in FIG. 1). In the presently preferred embodiment of the invention, the differential line drivers/receivers are model Nos. DS8921, manufactured by National Semiconductor. The drivers transmit a positive version of the data on one wire of the twisted-pair cable and the inverse of the data on the other wire of the twisted pair. This allows the data to be transmitted along cables up to 500 feet in length without the use of additional amplifiers.

As the user is operating the remote server computer, the remote computer may transmit commands which affect the operation of the mouse and keyboard. These include the mouse sensitivity, the keyboard repeat rate, activating one or more LEDs on the keyboard (such as the number lock, capital letter lock, etc.). The keyboard/mouse commands contained in a pod to pod packet transmitted from the remote computer are received on twisted-pair cable 72*b* by the differential line driver/receiver 88. The UART 84 converts the received serial keyboard/mouse commands into a parallel format and supplies the data to the CPU 80. The CPU 80 then generates the appropriate signals which are fed to the keyboard/mouse interface 82 and applied to the keyboard 62*b* and mouse 62*c*.

Video signals transmitted from the remote server computer are received on three sets of twisted-pair cables 72*f*, 72*g*, and 72*h* by a set of differential line receivers 90. The output signals produced by the differential line receivers 90 are supplied to a video amplifier 92. The output of the video amplifier is coupled to a sync extract circuit 94 which removes an embedded horizontal and vertical sync signal as well as a mode signal from the green, blue and red video signals respectively. The sync extract circuit 94 supplies the red, blue, and green analog video signals as well as the horizontal and vertical sync signals on separate leads to an onscreen programming circuit 99 that is described in further detail below. The onscreen programming circuit 99 feeds the video signals to a connector 96, which is coupled to the video monitor of the workstation by a conventional video cable 97. As will be described in further detail below, the horizontal and vertical sync signals are embedded into the green and blue color video signals in order to minimize the number of wires that extend between the workstation and the remote server computer as well as to reduce the complexity of the crosspoint switch.

The CPU 80 also reads a set of four monitor sense leads 95 to determine what type of monitor is connected to it. Monitor sense data is generated and transmitted in a pod to pod packet as shown in FIG. 2A. The remote computer receives the monitor data and supplies it to the remote computer in order to adjust its video signals accordingly.

In addition to transmitting and receiving keyboard and mouse signals from the remote computer, the pod 70 can communicate with the central crosspoint switch. Data to be transmitted to the central crosspoint switch are sent on a twisted pair cable 72*c* while data transmitted from the central crosspoint switch are received on a twisted pair cable 72*d*.

Commands sent between the pod 70 and the central crosspoint switch allow a user to connect the work station to another remote computer, allow the central crosspoint switch to interrogate the status of the pod, update the firmware of the pod, etc. using the packet structure shown in FIG. 2B as will be described below. When the user wishes to send a command to the central crosspoint switch, a special sequence of keystrokes is used. In the present embodiment of the invention, all commands are preceded by the "printscreen" key and end with the "enter" key. The CPU **80** parses the keyboard strokes for these keys and analyzes the keystrokes to determine the destination of the command. If the command is directed to the pod itself, no data packet is produced. If the command is directed to the remote computer, a pod to pod packet is generated and transmitted. If the command is directed to the central crosspoint switch, the CPU assembles a command packet that is transmitted to the central crosspoint switch on the twisted pair cable **72***c*.

A block diagram of a pod **76** that is coupled to the remote server computers is shown in FIG. **3**. The pod **76** includes a central processing unit (CPU) **120** that is coupled to a keyboard/mouse interface **134**. The keyboard/mouse interface **134** supplies signals to and receives signals from the server computer's keyboard and mouse connectors. The keyboard and mouse signals from the computer's keyboard and mouse connectors are read by the CPU **120** and assembled into a pod to pod packet in the same manner as the pod to pod packet described above and shown in FIG. 2A. The pod to pod packet produced by the CPU **120** is delivered to a QUAD UART **136** that transmits the packet serially over a lead **136***b* to a differential line driver **140**. The differential line driver drives a twisted-pair cable **74***a* that is coupled to the central crosspoint switch.

A pod to pod packet that is transmitted from a workstation is received on a twisted-pair cable **74***b* and supplied to differential line receiver **140**. The output signal of the differential line receiver is supplied to the QUAD UART **136** which converts the packet from a serial format to a parallel format. The CPU reads the packet and then transmits the received keyboard and mouse signals to the keyboard and mouse interface **134** where the signals are supplied to the remote computer's keyboard and mouse connectors in the same manner as if the keyboard and mouse were directly connected to the remote server computer. The particular format of the signals applied to the keyboard and mouse connectors may vary with the type of the remote computer. The CPU within the pod **76** is therefore programmed to translate the signals into their proper format.

Commands sent from the pod **76** to the central crosspoint switch allow the remote computer to interrogate the status of the pod, update the firmware of the pod etc. using the packet structure of FIG. **2B**. As with the user pod, all commands are preceded with the "printscreen" key and end with the "enter" key. The CPU **120** parses the keyboard strokes for these keys and analyzes the keystrokes to determine the destination of the command. If the command is directed to the pod **76**, no data packet is produced. If the command is directed to the workstation, a pod to pod packet is generated and transmitted. If the command is directed to the central crosspoint switch, the CPU assembles a command packet that is transmitted to the central crosspoint switch on a twisted pair cable **74***d*.

The signals from the remote computer's video port are supplied through a video cable **143** to a connector **144**. As will be described below, the red, green and blue analog video signals along with the horizontal and vertical sync signals are supplied to a sync combine circuit **146** that encodes the horizontal and vertical sync signals onto the green and blue analog video signals respectively. The current mode of the monitor (i.e., the correct polarity of the horizontal and vertical sync pulses) is encoded by the sync combine circuit **146** onto the red analog video signal. The output of the sync combine is supplied to an amplifier **148** that conditions the signals and supplies the video signal to three differential line drivers **140** that transmit the signals over three separate twisted-pair cables **74***f*, **74***g*, and **74***h* to the central crosspoint switch.

The monitor sense data received from a remote workstation is decoded by the CPU **120** and supplied to a set of monitor sense leads **147**. The remote computer receives the monitor sense data on these leads and adjusts its video signals for the particular monitor that is displaying the video signals.

The audio signals produced by the remote computer are supplied to a differential line driver **140** and are transmitted over a twisted-pair cable **74***c* to the central crosspoint switch.

FIG. **4** is a block diagram of the central crosspoint switch. The central switch **60** includes a master central processing unit (CPU) **150**, a number of input cards **152**, a number of switch cards **154** and a number of output cards **156**. Each of the input cards transmits signals to and receives signals from up to eight of the remotely located server computers while each of the output cards transmits to and receives signals from up to eight of the remotely located workstations. The master CPU **150** is coupled to each of the input cards **152**, the switch cards **154** and each of the output cards **156** by a digital bus **158**. Together the master CPU, input cards, switch cards and output cards are connected via a local area network.

Pod to pod packets are routed from an input card through the switch card to an output card and vice versa on a digital backplane **160**. The analog video and audio signals are transmitted between the input cards, the switch card **154** and the output cards **156** on a separate analog backplane **162**.

A block diagram of an input card **152** is shown in FIG. **5**. The output cards **156** are identical to the input cards except that the direction of the audio/video signals is reversed and therefore will not be discussed separately. The input card **152** includes its own CPU **170** that transmits and receives data from the master CPU **150**. Signals transmitted from the remote server computer are received by a set of differential line drivers/receivers **172***a–b*. Commands sent from the remote computer to the central crosspoint switch are received by an octal UART **173** where the commands are converted from a serial to a parallel format. The UART feeds the commands to the CPU **170** where they are interpreted and forwarded to the master CPU **150**.

To transmit data between the input, output and switch cards of the crosspoint switch, the data is packetized in the format shown in FIG. **2B** by the CPU of the card sending the packet. A packet begins with a unique character **112** that marks the beginning of the packet. A destination address **114** follows the start character. The address uniquely identifies one of the cards in the crosspoint switch. A byte **116** indicates the size of the packet while a byte **118** indicates the type of data included in the packet. A series of bytes **120** are the data to be transmitted from one card to another. Following the data, a byte **122** indicates the sending card's unique address. A checksum byte **124** follows the sender's address and a unique character **126** is sent as a trailer. The transmission of all data packets between the cards of the crosspoint switch is controlled by the master CPU **150**.

Returning to FIG. **5**, commands generated by the CPU **170** to be transmitted to the pod that is coupled to a remote server computer are transmitted on a lead **174***b* to a differential line driver 172. Pod to pod packets received from the central computer are routed through the input card on a lead **174***c* to the digital backplane **160**. Similarly, pod to pod packets transmitted from the remote workstation are received from the digital backplane, routed through the input card on a lead **174***d* and supplied to the differential line driver **172***a*.

In order to shield the video signals from the noise on the digital backplane, the video and audio signals transmitted from the remotely located server computer are routed on a separate analog backplane 162. The audio signals received from the remote computer are routed through the input card on a lead 174*e* and applied to the analog backplane 162. Video signals are received by the differential line receivers 172*a* and routed through the input card on leads 174*f*–*h* to the analog backplane.

In the present embodiment of the invention, each input card includes up to eight sets of differential line drivers/receivers 172*a*–172*f* (the remaining six driver/receivers not shown) to receive signals from up to eight remotely located server computers. The signals from each remotely located computer are routed through the input card to the digital and analog backplanes in the manner described above.

FIG. 6 is a block diagram of a switch card 154. The switch card includes its own central processing unit (CPU) 180. The CPU 180 transmits and receives signals from the master CPU 150 in order to control the position of a 16×16 digital crosspoint switch 182 and a 16×16 analog crosspoint switch 184 using a set of control leads 183. The digital crosspoint switch 182 connects the keyboard/mouse signals transmitted between a workstation and a remote server computer as well as audio signals generated by the remote server computer to the workstation. The analog crosspoint switch 184 transmits the video signals between a remote server computer and any of the workstations.

FIG. 7 shows how the digital backplane portion of the 32×32 crosspoint switch is configured using four switch cards 154*a*, 154*b*, 154*c* and 154*d* in order to transmit signals between 32 workstations and 32 remotely located server computers. The switch card 154*a* has sixteen input lines 186 that are coupled to sixteen remotely located server computers and sixteen output lines 188 that are coupled to sixteen workstations. The switch card 154*b* has sixteen input lines coupled to another sixteen remotely located server computers and sixteen output lines 194 that are coupled to each of the sixteen output lines 188 of the switch card 154*a*. The switch card 154*c* has sixteen input lines 198 that are coupled to the sixteen input lines 186 of the switch card 154*a*. The sixteen output lines 200 of the switch card 154*c* are coupled to another sixteen remotely located workstations. The switch card 154*d* has sixteen input lines 204 that are coupled to each of the sixteen input lines 192 of the switch card 154*b*. The sixteen output lines 206 of the switch card 154*d* are coupled to the sixteen output lines 200 of the switch card 154*c*. The analog backplane is constructed in a similar fashion as the digital backplane described above. As can be seen, the arrangement of the switch cards 154*a*, 154*b*, 154*c* and 154*d*, allows data from any one of thirty-two remotely located computers to be coupled to any one of thirty-two remotely located workstations.

A switching arrangement of the type shown in FIG. 7 is required for each signal that is to be transmitted between the remotely located server computer to a corresponding workstation. In the present embodiment of the invention, each workstation sends and receives pod to pod packets as well as receives audio and video signals from the remote computer. Therefore, for the 32×32 digital switch shown in FIG. 6, the digital backplane includes two sets of switches of the type shown in FIG. 7 and the analog backplane includes another four sets of switches for the video and audio signals.

In the presently preferred embodiment of the invention, the digital 16×16 switches 182 are implemented using a pair of 16×8 digital switches as shown in FIG. 8. Each 16×16 switch comprises switches 210 and 216. The switch 210 has sixteen input lines 212 and eight output lines 214. The switch 216 has sixteen input lines 218 that are coupled to each of the input lines 212, and eight output lines 220. In the presently preferred embodiment of the invention, each of the 16×8 switches 210 and 216 are part numbers CD22M34945Q, manufactured by Harris.

The analog backplane on which the video signals are transmitted is configured in the same fashion as the switch shown in FIG. 7. However, because of the greater bandwidth required, each 16×16 switch 184 is implemented using eight 8×4 analog switches model no. DG884DN, manufactured by Siliconix. As can be seen in FIG. 9, a 16×16 analog switch is implemented using switches 222, 224, 226 and 228 each having eight input lines and four output lines. The input lines of switches 222, 224, 226 and 228 are connected in parallel. A second set of switches 230, 232, 234 and 236, each having eight input lines and four output fines. The input fines of switches 230, 232, 234 and 236 are connected in parallel. The outputs of switch 230 are coupled in parallel with the outputs of switch 222, and the outputs of switch 232 are coupled in parallel with the outputs of switch 224. The outputs of switch 234 are coupled in parallel with the outputs of switch 226 and the outputs of switch 236 are coupled in parallel with the outputs of switch 228.

To minimize the number of wires that must extend from the remote computer to the workstation, the present invention encodes the horizontal and vertical sync signals onto the analog color video signals transmitted from the remote computer. FIGS. 10A–10C show the details of the sync combine circuit 146 (FIG. 3) that encodes the vertical and horizontal sync signals as well as the mode signal of the monitor. FIG. 10A shows a circuit that encodes the horizontal sync signal onto the green video signal produced by a remote computer. The circuit includes an exclusive or (XOR) gate 250 having a first input that receives the horizontal sync signal produced by the computer system. A resistor 252 and capacitor 254 are connected in a series between the first input of the XOR gate and ground. At the junction of the resistor 252 and the capacitor 254 are two series connected inverting gates 256 and 258. The output of the inverter 258 is supplied to a second input of the XOR gate 250.

The XOR gate 250 operates to encode the horizontal signal as a positively going pulse no matter what the normal state of the horizontal sync signal is. The voltage on the capacitor 254 is equal to the average valve of the horizontal sync signal. The output of the inverting gate 258 has a logic level equal to the non-active state of the horizontal sync signal. The output of the XOR gate 250 is coupled to an inverting input of an amplifier circuit 260. The non-inverting input of the amplifier 260 is connected to receive the green analog video signal. When the horizontal sync signal is in its normal state, the output of the amplifier 260 follows the green analog video signal. However, when the horizontal sync signal is activated, the active video is at zero volts and the amplifier 260 produces a negative going horizontal sync pulse.

FIG. 10B shows a circuit that encodes the vertical sync signal onto the blue analog video signal produced by the remote computer. The circuit comprises an exclusive or (XOR) gate 270, a resistor 272, capacitor 274 and a pair of inverters 276, 278 that are connected in the same way as the horizontal sync circuit shown in FIG. 10A and described above. The output of the XOR gate is always a positive going pulse when the vertical sync signal is activated. The output of the XOR gate is fed to the inverting input of an amplifier 280. When the vertical signal is in its normal state, the output of the amplifier 280 follows the blue analog video signal. However, when the vertical sync signal is activated, a negative going pulse, V-sync, is created by the amplifier.

FIG. 10C is an electronic circuit that encodes the mode of the video monitor. The mode refers to the polarity of the horizontal and vertical sync signals. Changes in the mode affect the size of the video display produced by a video monitor. To encode the mode of the video signal, the circuit shown in FIG. 10C is used. The circuit comprises two AND gates 284 and 286. The AND gate 284 has one input coupled to the output of the inverter 258 (shown in FIG. 10A). The AND gate 286 has one input coupled to the output of the inverter 278 (shown in FIG. 10B). The remaining inputs of the AND gates 284 and 286 are coupled to the output of the XOR gate 270 (shown in FIG. 10B) so that the mode signal is only encoded onto the red video signal when the vertical sync signal is activated.

The output of the AND gates 284 and 286 are coupled in series with a pair of resistors 290 and 292, respectively. The resistors 290 and 292 are coupled together at a common node 291. Connected between the node 291 and ground is a resistor 293. Each time the vertical sync signal is active, the AND gates 284 and 286 produce a voltage at the node 291 that is proportional to the mode of the video monitor. The proportional voltage is fed into the inverting input of an amplifier 294. The non-inverting input of the amplifier 294 is connected to receive the red analog video signal produced by the remote computer. When the vertical sync signal is in its normal state, the output signal of the comparator 294 follows the red analog video signal. However, when the vertical synchronize signal is activated, the mode signal is encoded on the red video signal.

After the video signals have been transmitted from the remote server computer and through the analog crosspoint switch to the remote workstation, the sync signals are extracted from the green and blue video signals. To extract the horizontal sync signal from the green video signal, the circuit shown in FIG. 11A is used. The green video signal is received by the pod at a differential receiver 90 that produces an output signal which is fed to a non-inverting input of a clipping amplifier 302. The output signal of the amplifier 302 is the green analog video signal that is fed to the video monitor. A resistor 306 is disposed between a non-inverting input of a comparator 304 to the output of the differential receiver 90. Connected between a non-inverting output of the comparator 304 and the non-inverting input is a feedback resistor 308. An inverting input of comparator 304 is tied to a constant reference voltage that is supplied by the voltage divider defined by resistors 310 and 312. When the output signal of the differential receiver 90 has a magnitude below the voltage provided at the inverting input of the comparator 304, the inverting output of amplifier 304 creates a positive going pulse. The positive going pulse is supplied to an input of an exclusive or (XOR) gate 314. Coupled to another input of the exclusive or gate 314 is the horizontal mode (H-mode) signal which is recovered from the red analog video signal as will be described below. The XOR gate 314 adjusts the polarity of the horizontal sync signal depending on the value of the H-mode signal.

The circuit required to extract the vertical sync signal from the blue video signal is the same as the circuit shown in FIG. 11A except that the exclusive or (XOR) gate receives the V-mode signal in order to adjust the polarity of the vertical sync signal.

To recover the video mode signal) the present invention utilizes the circuit shown in FIG. 11B. The red analog video signal is received at a pod by a differential line receiver 90 that produces the red analog video signal. The output of the differential line receiver 90 is coupled to the inverting inputs of a pair of comparators 320 and 324. The comparators 324 are gated by the output of a one shot 326 that is triggered by the rising edge of the vertical sync pulse so that the comparators only change state when the vertical sync signal is active. The noninverting input of comparator 324 is supplied with a reference voltage produced by a voltage divider that comprises a resistor 326 and a resistor 328. The inverting input of the comparator 320 is supplied with a constant voltage produced by a voltage divider that comprises a resistor 330 and a resistor 332.

A resistor 334 is placed between the output of comparator 320 and the inverting input of comparator 324. Finally, a resistor 336 is placed between the inverting input of comparator 320 and the inverting input of comparator 324.

The mode extract circuit produces two signals, H-mode and V-mode, having logic levels that are dependent on the magnitude of the mode signal encoded on the red video signal. If the magnitude of the mode signal is between 0 and −0.15 volts, the H-mode signal will be low and the V-mode signal will be low. When the mode signal has a magnitude between −0.15 and −0.29 volts, the H-mode signal will be high and the V-mode signal will remain low. The V-mode signal is high and the H-mode signal is low when the magnitude of the mode signal is between −0.29 volts and −0.49 volts. Both the H-mode and V-mode signals are high when the magnitude of the mode signal is less than −0.49 volts. As will be appreciated, the values given above will differ if different circuit components are used.

Once the video mode signal has been decoded from the red video signal, the values of H-mode and V-mode are used to adjust the polarity of the horizontal and vertical sync signals using the XOR gate shown in FIG. 11A.

As can be seen, the circuits shown in FIGS. 10A–10C and 11A, 11B reduce the number of wires that must extend between the remote server computer and the workstation by encoding the sync and mode signals onto the color video signals at a time when the signals are normally unused.

Having now described the components of the present invention, its operation is described. To connect a workstation to a remote computer, a user sends a command that causes the central crosspoint switch to couple the keyboard/mouse signals to one of the remote computers. As indicated above, commands that affect the operation of the crosspoint switch as inserted between "printscreen" and "enter" keystrokes. The pod connected to the workstation detects these keys and transmits a packet to the CPU on one of the output cards. The CPU then transmits the packet to the master CPU that validates the request and issues a command to the switch cards to set the position of the 16×16 digital and analog switches 182 and 184 (FIG. 6). Once the position of the switches has been set, the master CPU tells the computer pod 76 that the connection has occurred. The keyboard/mouse signals are then packetized and transmitted as pod to pod packets through the crosspoint switch. Video and audio signals from the remote computer are transmitted from the remote computer to the workstation.

As indicated above, the present invention provides the capability of allowing a user to send commands from a workstation to the central crosspoint switch in response to prompts that are displayed on the video monitor. The onscreen programming circuit 99 shown in FIG. 2 produces video signals that displays a menu of commands to be selected by the user. FIG. 12A is a circuit diagram of the onscreen programming circuit **99**. The circuit includes a set of tri-state buffers **352**, **354** and **356** that have their inputs connected to the red, green and blue video signals provided by the sync extract circuit **94** (shown in FIG. 2). When the tri-state buffers are energized, the red, green and blue video signals are passed to the video monitor. When the tri-state buffers **352**, **354** and **356** are in their high impedance state, the video signals are produced by an onscreen programming circuit **364**, as will be described.

The onscreen programming circuit **99** produces its own horizontal and vertical sync signals using a sync generator **358**. The horizontal and vertical sync signals produced are supplied to a switch **360** that selects either the sync signals produced by the internal sync generator **358** or the external horizontal and vertical sync signals recovered from the green and blue video signals transmitted from the remote computer. The switch **360** receives a signal on a lead **361** that is coupled to the CPU **80** (FIG. 2) that determines which set of horizontal and vertical sync signals are selected. The horizontal and vertical sync signals selected by the switch **360** are fed to the video monitor at the user's workstation. Also connected to the output of the switch **360** is a sync polarizer **362** that forces the polarity of the horizontal and vertical sync signals selected to be active low. The details of the sync polarizer **362** are shown in FIG. 12B.

The sync polarizer includes a pair of exclusive OR (XOR) gates **400** and **402**. The XOR gate **400** has one input connected directly to the sync signal to be polarized. A resistor **404** is connected between the sync signal and the other input of the XOR gate **400**. Connected between the second input of the XOR gate **400** and ground is a capacitor **406**. The voltage on the capacitor **406** is the average voltage of the sync signals. The output of the XOR gate **400** feeds an input of the XOR gate **402**. The other input of the XOR gate **402** is coupled to a logic high signal. The output of the XOR gate **402** will be a negative going pulse each time the sync signal is activated no matter what the normal state of the sync signal is.

The outputs of the sync polarizer **362** are coupled to a horizontal and vertical sync input of an onscreen processor **364**. The onscreen processor produces red, green and blue video signals that display one or more alphanumeric characters that are programmed in its internal video ROM memory. To dictate which characters are placed on the video screen, the CPU **80** generates serial I^2C interface signals on a pair of leads **363** and **365**. These signals are applied to the onscreen processor **364** which causes the processor to retrieve from an internal video RAM characters that are to be displayed on the video screen. The onscreen processor **364** provides two signals HBFK and HTONE that are supplied to an overlay control logic circuit **366**. Also supplied to the overlay control logic circuit are four signals from the CPU **80** of the user pod. These four signals are H Tone Enable, OSD Enable, System Video Enable and Transparent. The overlay control logic circuit **366** reads the value of these logic signals and either enables or disables a set of tri-state buffers **368**, **370** and **372** on the tri-state buffers **352**, **354** and **356**. These tri-state buffers **368**, **370** and **372** couple the outputs of the onscreen processor **364** to the leads that connect to the monitor's color inputs.

When the tri-state buffers **352**, **354** and **356** are in their high impedance state, and the tri-state buffers **368**, **370** and **372** are active, then the video screen will only display those signals produced by the onscreen processor. Conversely, if the tri-state buffers **368**, **370** and **372** are in their high impedance state and the tri-state buffers **352**, **354** and **356** are active then the monitor displays the video signals produced by the remote computer system. If both sets of tri-state buffers **368**, **370**, **372** and **352**, **354** and **356** are both active, then the monitor will display the video signals produced by both the onscreen processor and the remote computer system. The following is a table that defines the logic of the overlay control logic circuit **366**.

HTONE	HBFK	H TONE ENABLE	OSD ENABLE	SYS_VID_EN	TRANS-PARENT	DISPLAY
X	0	X	0	0	X	screen blank
X	X	X	0	1	X	state video displayed only
X	1	0	1	0	0	OSD displayed only
1	1	1	1	0	0	OSD with transparent characters, i.e., characters transparent, OSD windows solid
X	X	X	1	0	1	illegal state
0	1	0	1	1	0	active system video with solid OSD characters
1	1	1	1	1	0	active system video transparent OSD characters and solid OSD windows
1	1	0	1	1	1	active system video with opaque OSD characters and windows
1	1	1	1	1	1	active system video

-continued

HTONE	HBFK	H TONE ENABLE	OSD ENABLE	SYS_VID_EN	TRANS-PARENT	DISPLAY
						transparent OSD characters and opaque OSD windows

The construction of the overlay control logic circuit **366** given the above table is considered to be within the skill of an ordinary digital electronics engineer.

To activate the onscreen programming display, the user begins the escape sequence by pressing the "printscreen" key. The CPU within the user pod recognizes this key and produces a menu on the video screen. The user then selects one or more items from the menu by typing on the keyboard or moving the mouse. The CPU then interprets these mouse/keyboard inputs as commands that are to be transmitted to the central crosspoint switch. Once the user ends a command by activating the "enter" key, the CPU can generate one or more packets that are transmitted to the central crosspoint switch that enable the user to connect to a different computer, monitor the status of a different computer, etc.

As can be seen, the present invention allows a user to access any of thirty-two remotely located computers from a central workstation. The system operates apart from a network so that if the network fails, a user can still access each of the server computers. Furthermore, the pods act as translators between different keyboard/monitor types and different computers. Because all pod to pod packets have the same format, previously incompatible equipment can be easily coupled together.

While the preferred embodiment of the invention has been illustrated and described, it will be appreciated that various changes can be made therein without departing from the spirit and scope of the invention. For example, although the present invention is described with respect to connecting workstations to remotely located computers for the purposes of system administration, it will be appreciated that the invention also has further uses. For example, it may be desirable to locate expensive computer equipment away from relatively inexpensive terminals. Therefore, the present invention could be used in academic sessions where it is desirable to allow students to operate remotely located computers from one or more workstations. It is believed that the present invention has numerous applications where it is desirable to separate computing equipment from computer display and data input devices. Therefore, the scope of the invention is to be determined solely from the following claims.

What is claimed is:

1. A system for connecting a number of workstations of the type that include a keyboard, mouse and video monitor to a number of remote computer systems, comprising:

a plurality of first signal conditioning units coupled to the workstations for receiving electronic signals produced by the keyboard and mouse and for creating a serial data packet that includes the electronic signals;

a plurality of first communication links coupled to the first signal conditioning units for carrying the serial data packets;

a central crosspoint switch including a number of inputs and a number of outputs, said central crosspoint switch receiving the serial data packets from an input and routing the serial data packet to one or more of said outputs;

a plurality of second communication links coupled to the outputs of the central crosspoint switch; and

a plurality of second signal conditioning units coupled to the remote computer systems, for receiving the serial data packets transmitted on one of the plurality of second communication links switch and for supplying the data packets to a keyboard and mouse input of the remote computer.

2. The system of claim **1**, wherein the plurality of second signal conditioning units receive video signals produced by the remote computer systems and transmit the video signals to the central switch on one of the plurality of second communication links.

3. The system of claim **2**, wherein the video signals include a red, green and blue video signal as well as a horizontal and vertical sync signal, and wherein each of the second signal conditioning units includes an encoder circuit that encodes the horizontal and vertical sync signal onto the two of the red, green or blue video signals before the video signals are transmitted to the central switch.

4. The system of claim **3**, wherein the video signals include a mode signal that indicates a polarity of the horizontal and vertical sync signal, and wherein the encoder circuit encodes the mode signal onto one of the red, green or blue video signals before the video signals are transmitted to the central switch.

5. The system of claim **3**, wherein the first signal conditioning units include a decoder circuit for removing the horizontal and vertical sync signals from the red, green or blue video signals.

6. The system of claim **5**, wherein the decoder circuit removes the mode signal from the red, green or blue video signals.

7. The system of claim **6**, wherein the decoder circuit includes a circuit for adjusting the polarity for the horizontal and vertical sync signals based on the decoded mode signal.

8. The system of claim **7** further comprising an onscreen programming circuit that included in the first signal conditioning unit, the onscreen programming circuit producing video signals that are displayed by the video monitor.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 6,345,323 B1 Page 1 of 1
DATED : February 5, 2002
INVENTOR(S) : Danny L. Beasley et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 8,
Line 7, change "CD22M34945Q" to -- CD22M3494SQ --.

Signed and Sealed this

Nineteenth Day of November, 2002

Attest:

JAMES E. ROGAN

Attesting Officer

Director of the United States Patent and Trademark Office